Quantifying uncertainty in credit rating model development

Rebecca Zaalberg

Submitted to the University of Twente in partial
fulfillment of the requirements for the degree of Master
of Science in Applied Mathematics

August 24th, 2013

UNIVERSITY OF TWENTE.

Company supervisor:

Drs. J. Hommels

University supervisors:

Prof. dr. A. Bagchi

Dr. P.K. Mandal

Colophon

Title	Quantifying uncertainty in credit rating model development
Date	August 24th, 2013
On behalf of	Rabobank International – Quantitative Risk Analytics & University of Twente
Author	Rebecca Zaalberg
Supervisor Rabobank	Drs. J. Hommels
First supervisor University of Twente	Prof. Dr. A. Bagchi
Second supervisor University of Twente	Dr. P.K. Mandal

Abstract

Banks estimate the creditworthiness of their counterparties by using credit rating models. These models are developed internally, using statistical methods. The input for the final model are certain characteristics (factors) of a counterparty, based on which the model returns a credit rating for this counterparty.

During the model development, regression is performed on the factors to obtain the appropriate factor weights. In this thesis, we develop a method to quantify the uncertainty in these estimated weights.

After the weights are fully determined, the performance of the model is checked, and – in particular – its discriminatory power. For this, the measure *powerstat* is used. Also for this measure we provide a method to quantify its uncertainty.

Both methods are tested numerically and examples are generated for illustration purposes. Where possible, datasets from real model developments are used.

Acknowledgements

This thesis concludes six months of research at the Quantitative Risk Analytics (QRA) department at Rabobank International. I would like to express my gratitude towards a number of people that helped me with my research.

First of all, I thank drs. Jasper Hommels for his daily supervision. His creative solutions and practical insights have greatly improved my research. Also, Jasper's enthusiasm was very motivating throughout this whole internship. I also thank the other members of the QRA team for helping me when necessary.

From the university, I thank prof. dr. Arun Bagchi and dr. Pranab Mandal for our fruitful discussions and their constructive feedback on my thesis. Also, I thank dr. Klaas Poortema for being part of the graduation committee.

Last but not least, I thank my family and friends, and Rob in particular, for their support and for motivating me when necessary.

Rebecca Zaalberg
Utrecht, August 2013

Contents

1 Introduction

Currently, we are experiencing an economic crisis, following and caused by the banking crisis that started in 2008. This has prompted banks all over the world to pay more attention to their risk management: the managing of the risks a bank encounters on a daily basis.

In order to identify the risks a bank faces, we have to look at its core functions. One of these core functions is granting loans. The party on the receiving end of this loan is called the *counterparty*. It can happen that the counterparty fails to meet its obligations following from the loan. The risk that this situation occurs is called *credit risk* (Basel Committee on Banking Supervision, 2000).

Credit risk is the most important risk for a bank (Bessis, 2011), so it is of vital importance for banks to properly manage it. This is accomplished by maintaining a stringent acceptance policy, setting adequate prices for their financial products, and keeping a large enough capital buffer. For these tasks banks make use of credit ratings in order to rate their counterparties. These ratings reflect the *creditworthiness* of a counterparty: how likely is it that the counterparty will meet its financial obligations in full and on time (Standard & Poor's, 2011). A counterparty with a strong capacity of meeting its financial commitments will be given a high rating, whereas a counterparty that is on the verge of default will get a low rating.

Two types of ratings can be recognized: external and internal ratings. External ratings are ratings given out by independent rating agencies, such as Standard & Poor's (S&P), Moody's, and Fitch. These ratings are publicly available. On the other hand, internal ratings are only given out and used *within* a bank. In Rabobank's case, it rates its counterparties internally and uses these ratings internally as well. Banks need internal ratings, as external ratings are only available for large counterparties (Hull, 2010). Also, the Basel Committee of Banking Supervision (the international supervisory authority for financial institutions) encourages banks to use internal models (Basel Committee on Banking Supervision, 2006).

In order to have some more background for understanding the problems that will be addressed in this thesis, we first need to explain more on how the internal ratings are developed. This is done in 1.1. Then, in 1.2 we describe the two problems that will be addressed in this thesis. Finally, in 1.3 we give an overview of the outline of this thesis.

1.1 Internal rating models

In order to manage their credit risk in a sensible manner, Rabobank gives each of their (potential) counterparties a rating from one of the 21 rating classes it identifies. These classes are ordered from high to low creditworthiness.

If a counterparty has a very low creditworthiness, it is likely that a default will occur. A default is defined by Basel II (the methodology for setting capital requirements for financial institutions) as the occurrence of one of the following events (Basel Committee on Banking Supervision, 2006):

- The bank considers that the obligor is unlikely to pay its credit obligations to the banking group in full, without recourse by the bank to actions such as realising security (if held).
- The obligor is past due more than 90 days on any material credit obligation to the banking group. Overdrafts will be considered as being past due once the customer has breached an advised limit or been advised of a limit smaller than current outstandings.

Rabobank uses this definition for defaults as well. It is used to determine whether a counterparty is in default.

It is very important to a bank to correctly estimate the probability that a counterparty will default within the next year. This probability is called the *Probability of Default* (PD). The PD is used for the following important applications:

- Determining whether a potential counterparty should be accepted as a client,
- Pricing the financial products for this counterparty,
- Computing the capital buffer that should be held in order to remain solvable in adverse situations.

In order to estimate the PD for these applications, the internal ratings are used. Using a predetermined mapping, the ratings are simply mapped to a PD.

In order to give each counterparty an internal rating, Rabobank (as any other bank) uses rating models. These models use certain characteristics of the counterparty (e.g. solvency ratio, history with Rabobank) as input factors. The output of a rating model is the internal rating for the counterparty.

Two main types of rating models can be distinguished over the years (Sobehart & Keenan, 2007). The *fundamental* models are based on Merton's model (Merton, 1974). This model sees the counterparty's equity as an option on its assets. As Merton's model was based on some unrealistic assumptions, extensions followed quickly. The pending implementation of Basel II in the mid-nineties sped up the model refinements even further (Altman, 2006). The models that are currently used within Rabobank belong to the other type of rating models, the *statistical* models: those that determine the relationship between defaults and market information. Rabobank's credit rating models can also be seen as an extension of the Altman Z-score (Altman, 1968): a score for the creditworthiness based on financial ratios.

Rabobank develops its rating models internally. This is done using historical data (if available) for creating a statistical model, combined with expert input. To create a statistical model, two types of historical data are required: information on the characteristics of the counterparties, and information on the creditworthiness of the counterparties. The creditworthiness information can be either occurrences of defaults (or of non-defaults) or external ratings. As there is generally limited data available, experts are involved in the modelling process to adjust the statistical results according to their expertise.

The rating models are reviewed on a regular basis. If a model's performance is deemed insufficient, Rabobank redevelops the model including more recent data in order to create a better model (Rabobank, 2010).

1.2 Problem statement

A rating model is developed using historical data combined with expert input. First, a statistical model is developed based on the historical dataset. Then, a panel of experts checks this model and may ask for some modifications. After finalizing the model, its discriminatory power (proficiency in discriminating between groups of different creditworthiness) is tested using the development dataset, or – if new data is available – a more recent dataset. We will go into further depth on the development of a rating model in a later chapter.

These operations involve statistical estimations on finite datasets, so there is some uncertainty in these estimates. Quantifying this uncertainty can improve the model development methodology and the quality of the estimates. The problem of quantifying the uncertainty of statistical estimates appears in two stages of the model development process: during the estimation of the model, and during the testing of the discriminatory power of the model.

Problem 1
An important part of developing a rating model, is determining the weights for the factors that explain the creditworthiness of a counterparty. The weights that are obtained using regression, are presented to a panel of experts. It often occurs that these experts suggest an alteration in one or more of the weights. As the weights are estimated on a dataset, there might indeed be some freedom in changing these weights without affecting the statistical power of the estimate. But how much change is allowed? To answer this, we need to quantify the uncertainty in the weight estimates.

Problem 2
A rating model can be judged on a number of characteristics. One of those is the discriminatory power of the model. This is measured by the so-called *powerstat*. The higher the powerstat, the more discriminatory power the model has. By introducing a new factor, for example, the powerstat may be slightly increased. But is this increase significant? If this is not the case, we would not even bother increasing the complexity of the model. The uncertainty in the measured powerstat should therefore be quantified. Then we know how much increase in the powerstat indicates a significant improvement of the model.

These two problems are addressed in this thesis.

1.3 Outline

In this thesis we investigate methods to quantify the uncertainty in two stages of the rating model development. For this, it is essential to be familiar with the rating model development process. Therefore, the (re)development of a PD model is explained step by step to give a better understanding of the modelling process. This is done in Chapter 2.

In Chapter 3 we introduce a method to solve the first problem, as stated in 1.2. That is, a method is described that yields the interval over which a weight is allowed to be changed. In the following chapter (Chapter 4) the method is tested numerically and examples of applications of this method are given.

The second problem is addressed in Chapter 5. In this chapter a confidence interval is created for the powerstat by relating this measure to another measure: the divergence. Numerical applications of this method is given in Chapter 6.

Finally, in Chapter 7 the conclusions are given, together with suggestions for further research.

2 Rating model development

This chapter is devoted to giving a detailed insight in the process of (re)developing a credit rating model. This knowledge is needed as a basis for the following chapters.

The core activity of a bank is giving out loans. Therefore, one of the most important risks a bank has to protect itself against is credit risk: "the potential that a bank borrower or counterparty will fail to meet its obligations in accordance with agreed terms" (Basel Committee on Banking Supervision, 2000). For Rabobank (as any other bank) it is therefore of vital importance to model the creditworthiness of its counterparties as well as possible.

A part of this is modelling the Probability of Default (PD). This is an important input parameter for determining the capital buffer that the bank needs to cover the risk it takes and the losses it expects. Also, the PD is used to determine if a potential counterparty can be accepted as a client and which price should be asked for the financial products. Based on specific group characteristics – such as region, size, and industry – the total portfolio of the bank can be split into several portfolios. For each of these portfolios, a PD model can then be created.

Regarding the international financial crisis of the last few years, a model redevelopment was needed for the model estimating the PD of commercial banks. Also, recently the PD model for Small and Medium Enterprises (SME) in Poland has been developed. This was done, following a special request of the Polish regulator. These two PD models and their development datasets will be used for illustration purposes throughout this chapter.

In this chapter, we will first discuss the requirements and the approach for a rating model (re)development (section 2.1). The four steps of this approach will be discussed separately in the sections 2.2 to 2.5.

2.1 Modelling starting points

A rating model is developed based on expert input and – if available – historical data. This data contains *observations*. An observation is defined as a snapshot of all information available of a counterparty at some time in the past. This includes the qualitative and quantitative information, and the external creditworthiness information. These first two types of information contain the characteristics of a counterparty. Throughout a rating model development these characteristics are used as the explanatory factors of the creditworthiness. Since the quantitative factors are all financial characteristics, they will be referred to as *financial* factors.

The goal of a rating model development is developing a model with financial and qualitative factors as input, that ranks counterparties in terms of creditworthiness. A rating model consists of two major layers: the ranking and the calibration (Rabobank, 2010). During the ranking stage, the score for a counterparty is determined as a linear combination of the factor scores. We will call this the *model score*. The weights of the factors are computed during the model development. The next stage is the calibration, in which the model score for a counterparty is first mapped to a PD. Subsequently this PD is mapped to a rating

category. Often, an estimate for the PD can be computed directly from the model score. For those models the calibration only involves a mapping from the PD to a rating category.

New and redeveloped rating models should satisfy a number of requirements, as set by the regulator or Rabobank itself. We elaborate on these requirements in subsection 2.1.1.

The type of creditworthiness information of the counterparties considered during the (re)development process depends on the number of defaults available in the dataset. It determines the modelling approach that will be used. We will focus on the two modelling approaches that use statistical analysis: the Good-Bad approach and the Shadow-Bond approach. These approaches differ in the statistical methods that are used for the model development. More on the modelling approaches is found in subsection 2.1.2.

The process of a rating model (re)development can be divided into four steps. These are introduced in subsection 2.1.3.

2.1.1 Requirements

The new or redeveloped model should comply with both internal and external requirements. The external requirements are set by Basel II (Basel Committee on Banking Supervision, 2006) and De Nederlandsche Bank. These, and internal Rabobank requirements, are summarised in "General checklist PD models" (Rabobank, 2006). The most important ones are listed below.

- Rating models should be based on historical experience and empirical evidence. The historical data should be representative for the portfolio to be rated. It is emphasized that neither historical data, nor expert judgement alone is a sufficient basis for the rating model development. A rating model should therefore incorporate both historical data and expert judgement.
- The historical data on which the model is based should have a length of at least one business cycle, which is approximately five years.
- The model needs to be accurate: its outcomes should agree with external ratings.
- The model and its outcomes have to be robust. That is, the model should also perform well on other datasets than the development dataset and the estimated factor weights of the model should be stable.
- The model has to be intuitive, so it should agree with the expert judgement.

2.1.2 Good-Bad approach and Shadow-Bond approach

A rating model is developed using expert input and – if available – historical observations. Preferably, the model should be based on data containing both default and non-default observations. In combination with expert input, statistical analysis is performed on the factors of both the defaulted and the non-defaulted counterparties in order to create a model based on those factors that discriminate best between the defaulted and non-defaulted counterparties. This is also known as the Good-Bad approach (Rabobank, 2010).

In order to obtain a statistically strong model using the Good-Bad approach, it is necessary to have enough default and non-default observations. This requirement cannot always be met, however. This is the case for so-called *low default portfolios* – portfolios that hardly ever

experience a default. If the Good-Bad approach would be used on a low default portfolio, the few default observations would have an unjustifiable large influence in the model development.

If there are too few observations to use the Good-Bad approach, we might choose to rely on expert judgement entirely for developing the rating model. But there is a middle way: the Shadow-Bond approach. A model that is built using this approach can be described as a "statistical model built to mimic external ratings" (Standard Chartered, 2008). So instead of basing the model on defaults and non-defaults, external ratings are used as the basis of the model development under the Shadow-Bond approach. An immediate natural question is then: why does Rabobank not use the external ratings directly, instead of developing a model that mimics them? We will give the main reasons for using the Shadow-Bond approach.

- In general, external rating agencies do not share how they create their ratings. Therefore, developing an internal model gives more insight in how a rating is obtained.
- An internal rating model allows for modifications to the rating due to expert input. This way, the bank's own opinion on a certain counterparty can be incorporated in the rating for this counterparty.
- Not all companies are rated by external rating agencies. An internal model can provide ratings for these counterparties as well.
- Regulation requires banks to rely less on external ratings and encourages the use of internal rating models (Basel Committee on Banking Supervision, 2006).

Whereas the observations used for the Good-Bad approach are split in two groups (default and non-default), the observations for the Shadow-Bond approach are divided over multiple groups. The number of groups corresponds to the number of external rating categories. Since these categories are ordered, statistical conclusions can be drawn on the relationship between creditworthiness and explanatory factors.

Take for example a qualitative factor with scores on a scale of 0 to 10. Suppose that counterparties rated CCC, BB, and AAA, have an average factor score of 2, 6, and 10, respectively. This suggests that there is a positive relation between this qualitative factor and the creditworthiness of a counterparty. This factor is therefore likely to be included in the model.

From this example we can also see that dividing the observations over more than two groups creates a finer granularity and might therefore make it easier to discover factors with some explanatory power.

But the Shadow-Bond approach also has its disadvantages. Whereas the Good-Bad approach is based on observed defaults – which is direct creditworthiness information – the Shadow-Bond approach uses external ratings that are supposed to represent the creditworthiness of a performing counterparty. This can be seen as "indirect" creditworthiness information and the dependent variable is thus of lesser quality.

Furthermore, this approach is highly dependent on the quality of the external ratings. Suppose, for example, that the external ratings would be overly optimistic for very high and

very low rated counterparties, but overly pessimistic for counterparties with medium ratings. In that case, the ratings given by the model would show the same behaviour.

Also, external ratings may be missing for some counterparties in the portfolio. This requires more representativeness adjustments to the set of observations (Rabobank, 2010).

For the reasons mentioned above, the Good-Bad approach is preferred over the Shadow-Bond approach. However, if there is not enough data available to use the Good-Bad approach, the Shadow-Bond approach provides a useful alternative, without completely relying on expert judgement.

The commercial banks PD model will be redeveloped following the Shadow-Bond approach, since the commercial banks' portfolio is a low default portfolio. The number of observations available for a model development based on the Shadow-Bond approach should be more than 100 (Rabobank, 2010). This is the case for the development dataset of the commercial banks model redevelopment. On the other hand, the Poland SME portfolio contains enough defaults for the Poland SME PD model to be developed following the Good-Bad approach. That is, the number of observations is at least 600 with 60 "bads".

2.1.3 Four modelling steps

Following Rabobank's guidelines for (re)developing a rating model (Rabobank, 2010), four steps can be distinguished in the process of developing a credit rating model. See Figure 1.

Figure 1: The four steps of the rating model development.

The first step in redeveloping a new rating model, is to gather as much data as possible. Also, the data is cleaned, that means that unusable data is removed. Since the resulting dataset (the development dataset) is not the same as the portfolio that it is supposed to represent, the representativeness of the development dataset needs to be checked by comparing it with the portfolio.

After that, the Single Factor Analysis (SFA) is performed. From this analysis, we determine the stand-alone explanatory power of each factor. The goal of the SFA is to find the most important factors for explaining the creditworthiness of the counterparties. Also, the factors are transformed in preparation for the next modelling step.

This next step is the Multi-Factor Analysis (MFA). The input of the MFA are the factors that were found to have the highest stand-alone explanatory power from the SFA. Using a regression method, the factor weights are determined. The choice of regression method depends on the available creditworthiness information: if observations are marked as either "default" or "non-default" (according to the Good-Bad approach), the logistic regression method is used. On the other hand, if external ratings are available as the creditworthiness

information (Shadow-Bond approach), linear regression is used to determine the factor weights. If the factor weights are determined, the model score can be computed for each counterparty as the weighted sum of the factor scores.

Finally, in the last step of the model development, the model scores are linked to a Rabobank Risk Rating. After that, the model is tested regarding its robustness, its improvement compared to the old model (if an old model exists), and its compliance with the requirements. Also, a user acceptance test takes place. During this final test end-users of the model will use the model and give feedback on its performance. As the first three steps are most important for the rest of this thesis, we will discuss the final step of the model development only briefly.

2.2 Dataset creation

We mentioned before that the model is developed based on a dataset containing observations. An observation is a snapshot of all information available of a bank at a certain time. This includes the qualitative, financial and creditworthiness information.

Before the data can be used for the model redevelopment, some data cleaning has to be done in order to create reliable observations. This is done according to the procedure as described in Rabobank's guidelines for rating model development (Rabobank, 2010). Furthermore, it should be checked whether the cleaned dataset is representative for the whole portfolio. For example, if 90% of the counterparties in the portfolio of commercial banks is located in an industrialized country, a development dataset for the commercial banks PD model with only 30% of the observations located in industrialized countries would not be representative.

2.2.1 Data cleaning

The data cleaning process can be divided into two parts: the initial data cleaning and the factor data cleaning. The initial cleaning can be done without even looking at the qualitative and financial information. During the *factor* data cleaning, the qualitative information and the financial information are cleaned, since these types of information will be used as explanatory factors for the creditworthiness of an observation.

During the initial cleaning, observations are removed if there is something wrong with their non-factor information. That is, observations can be removed for one or more of the following reasons:

- The observation is never approved by the credit committee and is therefore not a valid observation.
- Within 30 days, the observation is followed by another observation of the same counterparty. The two observations are very likely to have exactly the same factor information and are therefore essentially the same observation. In order not to double the weight of this observation, one (the oldest) should be removed. The limit of 30 days is set by experts.
- The counterparty observed enjoys parental support, which means that if the counterparty would have difficulties meeting its financial obligations, it can count on

the parent company for help. As the model is developed to rate stand-alone counterparties, counterparties with parental support should be removed from the dataset.
- The observation needs to be recent enough. On the other hand, bank supervisors require rating models to be built on at least five years of data. Therefore, a time threshold should be set (by experts) and observations that are older than this threshold should be removed.

Depending on the model to be developed and the dataset, additional restrictions to the observations can be made. Only if an observation matches all of the restrictions, it passes the initial cleaning process.

We then focus our attention on the factor information. For the qualitative information, the cleaning is mainly restricted to correcting typographical errors. Since the qualitative factors' scores are discrete, these errors are easily detected. Also for the financial factors the random errors should be corrected, although finding these errors is harder, for these factors' scores are continuous. After this, the outliers are detected in the financial data and the observations with outliers are removed.

Often the financial data contains missing values. In order to clean the missing values, the experts first split the financial factors into two categories: the regular factors and the exceptional factors. The regular factors are assumed to be filled for all observations. A missing value for a regular factor is therefore filled by interpolation or extrapolation if possible, and otherwise by the median of the factor. However, if more than 50% of an observation's regular factors is missing, the observation is removed. The exceptional factors are factors of which experts believe that they will not be available for all observations. These factors will only be filled incidentally and missing values are therefore replaced by zeros.

The financial factor information consists of the financial values as published by the counterparties. However, in predicting a counterparty's creditworthiness, some financial ratios may be more informative than financial values. Consider for example the ratio "interest expenses over liabilities". It makes sense that if a company has more liabilities, the interest expenses increase. But if the ratio "interest expenses over liabilities" increases, this may signal a negative effect on the creditworthiness. Therefore, after cleaning the financial information some useful ratios are computed. These ratios are also used as explanatory variables during the next steps of the model development. But first, a final cleaning step is required: if the denominator of a ratio value is zero, it is filled with the median of the ratio after the factor transformation (see 2.3.2).

2.2.2 Representativeness

Since the development dataset is only a part of all the data of the portfolio that is to be studied, the representativeness of the development dataset needs to be checked by comparing it with the portfolio.

For the commercial banks PD model redevelopment, we compare the development dataset with the Rabobank's portfolio of commercial banks. There are two geographical

characteristics for which the dataset is checked: region of risk and industrialization of the country of risk.

For the first breakdown, the commercial banks' portfolio observations and observations from the development dataset for the commercial banks PD model are split according to region. This breakdown recognizes five regions: Asia, Australia, Europe, Latin America, and the United States. It follows that the observations of the development dataset are spread over the regions in a similar way as the observations of the portfolio. This suggests that – with regards to the regions of risk – the development dataset is representative for the commercial banks' portfolio.

Another way of checking the representativeness of the development dataset, is splitting it in two groups: those in Industrialized Countries (IC) and those in Emerging Markets or Developing Countries (EM/DC). Also from this breakdown it follows that the development dataset for the commercial banks PD model is representative for the commercial banks' portfolio.

Similarly, other breakdowns can be used to check the representativeness of the development dataset. The choice of breakdowns depends on the portfolio. For example, the two breakdowns for the commercial banks' portfolio as illustrated above are not informative for the Poland SME portfolio. All observations in this portfolio and in the development dataset for the Poland SME model are in Poland. Therefore, for both the region breakdown and the industrialization breakdown, 100% of the observations of both the portfolio and the development sample would be labelled "Europe" and "EM/DC", respectively. For example, a breakdown based on the size of the company may be more informative for this portfolio.

2.3 Single Factor Analysis

Throughout the rest of this chapter we will label both the qualitative and financial factors and the financial ratios as "factor", for simplicity.

During the Single Factor Analysis (SFA) we test the explanatory power of each factor (qualitative and financial) on a stand-alone basis (Rabobank, 2010). That is, for each factor we check the discriminatory power of the model that is only based on that specific factor. Also, the factors that are to be included in the model, should be intuitive. So if experts expect a factor to have a positive effect on the creditworthiness (for example, the size of a counterparty), this should be reflected by the data. If this is not the case, the factor is left out of the model development. Furthermore, during the SFA the factors are prepared for the Multi-Factor Analysis (MFA). This preparation includes a transformation to diminish the effect of outliers and to give the factor scores the same range: from 0 to 10.

The goal of the SFA is to find a short list of important factors for explaining the creditworthiness of the counterparties. One may suggest using the reputable Principal Component Analysis (PCA) (Jolliffe, 2002) for finding these most important factors. This is not done, however. There are two main reasons for not using PCA during the SFA. Firstly, the SFA relies heavily on expert opinion. Based on tests of the factors on a stand-alone basis, the experts decide whether a factor should be included in the MFA. Using PCA does not yield

insightful results for the experts. Secondly, PCA already considers inter-factor relations. It delivers the combinations of factors that have most predictive power. But since the factors should be judged on a stand-alone basis, PCA cannot be used during the SFA.

Throughout the rest of this chapter we will use four factors to illustrate the methods for model development. Two of these factors stem from the commercial banks model redevelopment (one qualitative and one financial) and the other two from the Poland SME model development (one qualitative and one financial). See Table 1 for these factors.

Factor	Model (re)development	Financial/Qualitative?
Total assets ratio	Commercial banks	Financial
Market risk exposure	Commercial banks	Qualitative
Debt service coverage ratio	Poland SME	Financial
History with Rabobank	Poland SME	Qualitative

Table 1: Four factors from two different model (re)developments that will be used for illustration purposes.

2.3.1 Discriminatory power

The discriminatory power of a factor (or model) is the factor's (or model's) ability to discriminate between different groups of creditworthiness. Statistics for measuring a factor's (or model's) discriminatory power are for example the Gini coefficient (Gini, 1912), the accuracy ratio (Engelmann, Hayden, & Tasche, 2003), and the area under the receiver operating curve (Sobehart & Keenan, 2007). But within Rabobank the *powerstat* is used, which is quite similar to the Gini coefficient and the accuracy ratio. We will therefore use the powerstat as explained in Rabobank's guidelines (Rabobank, 2010) to measure the discriminatory power of a factor or model.

In this chapter, we look at the powerstat from a practical point of view. The computation of the powerstat is therefore explained algorithmically. A more theoretical discussion of the powerstat can be found in Chapter 5. Also, during the SFA we want to compute the powerstat for a *factor*. We will therefore explain the computation of the powerstat using the factor scores. For computing the powerstat of a model, the model scores should be used instead.

The powerstat compares the scores of a factor to the scores of a model with perfect discriminatory power. The closer the factor is to having perfect discriminatory power, the higher the powerstat. So if a factor has a high powerstat, its information is useful for estimating the credit rating. On the other hand, a factor that has the same score for all observations, has no discriminatory power at all and therefore has a powerstat value of zero.

The powerstat is computed in the following way. First, all observations are ordered according to the factor score. When using the Good-Bad approach, a score is fixed and subsequently the proportion of bad observations with a lower or equal factor score (number of bad observations with a lower or equal factor score divided by total number of bad observations) is plotted against the proportion of *all* observations with a lower or equal factor score (number of observations with a lower or equal factor score divided by total number of observations). This is done for all possible scores, thus creating the *power curve*.

When the Shadow-Bond approach is used, there are no "bad" observations. The power curve is therefore constructed in a slightly different manner. For a fixed score, the sum of the PDs of observations with a lower or equal factor score is determined. This is divided by the sum of the PDs of all observations, thus determining the weighted proportion. This proportion is then plotted against the proportion of all observations with a lower or equal factor score than the fixed score, just as for the Good-Bad approach. Again, this is done for all possible scores, thus creating the power curve.

To give some illustration, we consider the following example. Suppose we have a dataset of five observations. Of each observation we know the score for a certain factor, whether it is a default or a non-default, and also the PD corresponding to its external rating is known. Therefore, both the Good-Bad approach and the Shadow-Bond approach can be used. See Table 2.

Observation	Factor score	Good (0) or Bad (1)	PD
1	2	1	0.2
2	3	0	0.05
3	4	0	0.2
4	7	1	0.05
5	8	0	0.1

Table 2: Five fictional observations that are used to illustrate the computation of the powerstat under both the Good-Bad approach as well as the Shadow-Bond approach.

We see that 60% of all observations have a factor score lower or equal to 4. Of the bad observations, 50% has a factor score that is lower or equal to 4. Therefore, the point (0.6,0.5) lies on the power curve (blue) in Figure 2. The rest of this power curve is created similarly.

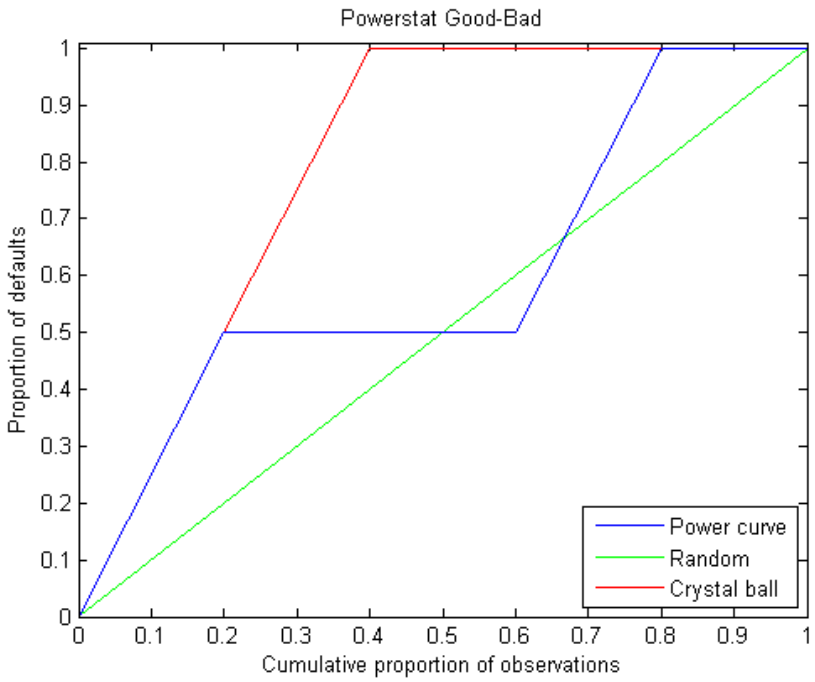

Figure 2: The power curve, the random model curve, and the crystal ball model curve for the data of Table 2 under the Good-Bad approach.

For the 60% of the observations with a factor score lower or equal to 4, the sum of their PDs is 0.45. The sum of all PDs in this dataset is 0.60, so the weighted proportion of observations with a factor score lower or equal than 4 is 0.45/0.60=0.75. Therefore, the point (0.6,0.75) lies on the power curve (blue) in Figure 3. The rest of this power curve is created similarly.

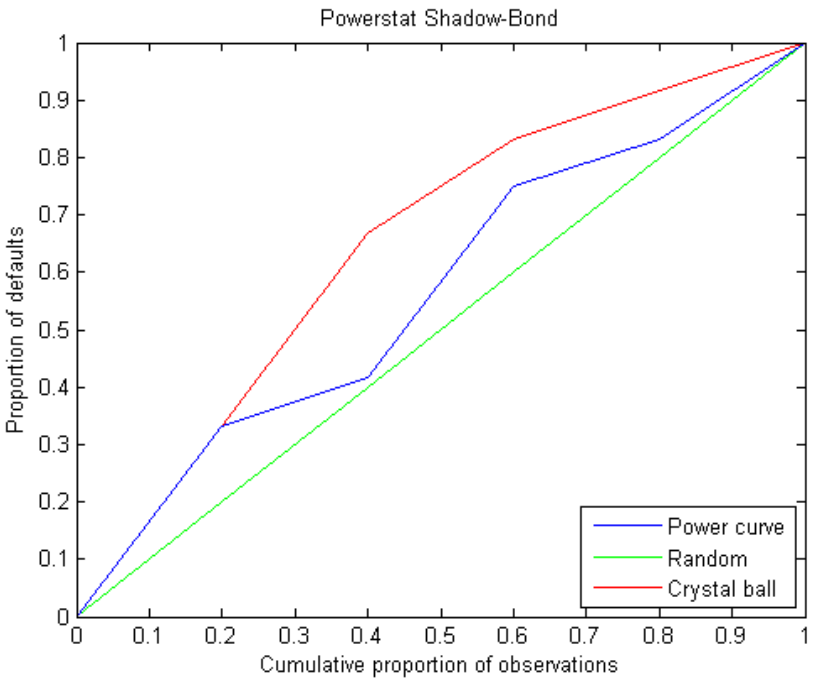

Figure 3: The power curve, the random model curve, and the crystal ball model curve for the data of Table 2 under the Shadow-Bond approach.

In addition to the power curve, also two other curves are shown in Figure 2 and Figure 3. One corresponds to the crystal ball model and the other to the random model. The crystal ball model is the model that the factor scores would follow if they had perfect discriminatory power. Using this model would be like having a crystal ball in which we could see the future (of the creditworthiness of the counterparties). So if the Good-Bad approach is used, the worst factor scores would be allocated to the defaulted observations. Indeed, we see in Figure 2 from the curve for the crystal ball model (red) that the two observations with the lowest factor scores are the only two bad observations. The curve consists of two parts: a part with a steep incline representing the bad observations, and a horizontal line representing the good observations. For the Shadow-Bond approach, if the factor scores follow the crystal ball model, the ordering of the factor scores corresponds exactly to the inverse ordering of the PDs.

The random model is the model that the factor scores would follow if they had no discriminatory power at all. This model randomly allocates scores to the observations. Therefore, the score of an observation does not depend on the creditworthiness of the observation. The slope of the curve of the random model is therefore constant, yielding a straight line. In both Figure 2 and Figure 3 the curve of the random model is the straight, green line.

The powerstat is then computed using these three curves. It is the ratio of two areas: the one between the power curve and curve of the random model and the one between curve of the crystal ball and curve of the random model. The powerstat therefore reflects how close the power curve lies to the curve of the crystal ball model. If the power curve is identical to the curve of the crystal ball model, the powerstat reaches its maximum value of 1.

For the commercial banks model redevelopment we specifically follow the financial factor "total assets ratio" and the qualitative factor "market risk exposure". For the Poland SME model development, the two factors "debt service coverage ratio" and "history with Rabobank" are followed. The powerstat for each of these for factors is given in Table 3.

Factor	Model (re)development	Powerstat
Total assets ratio	Commercial banks	0.534
Market risk exposure	Commercial banks	0.527
Debt service coverage ratio	Poland SME	0.357
History with Rabobank	Poland SME	0.347

Table 3: The powerstat values for four explanatory factors.

In general, the powerstat for a single factor is deemed "high" if it is above 0.30. Since this is the case for these four factors, they are very likely to be included in the next step of the model development process: the MFA.

We can also see that the powerstat values for the Shadow-Bond approach are higher than those for the Good-Bad approach. This is generally the case. Therefore, the powerstat for the Shadow-Bond approach should be interpreted differently than the powerstat for the Good-Bad approach. More on this difference in powerstat values will be discussed in 5.4.2.

2.3.2 Factor transformation

After selecting the factors that are deemed most important for explaining the creditworthiness of a counterparty, we may want to directly continue with the MFA. Theoretically, this can be done, but in order for the MFA to yield better and more insightful results, a factor transformation is required first. The reason for using this transformation is threefold (Rabobank, 2010):

- Outliers in the data can have an unjustifiable large influence on the regression outcomes. Applying a transformation on the factors before the regression can diminish their effect.
- Through transformation all factors can be given the same range. When regression is performed on these transformed factors, the resulting factor weights can be compared more intuitively.
- Some factors initially have a score distribution that is inversely related to the creditworthiness. So for those factors, a higher score would correspond to a lower creditworthiness. The final model is required to be intuitive: higher model scores should belong to counterparties with higher creditworthiness. To be able to easily check this, all factors should be positively related to the creditworthiness as well. Therefore, a transformation is required for those factors that are inversely related to the creditworthiness.

These three goals of the transformation can be achieved by using an S-shaped function. As the logistic function has such a shape, logistic transformations are used for transforming the factors. See Figure 4 for the logistic transformation function.

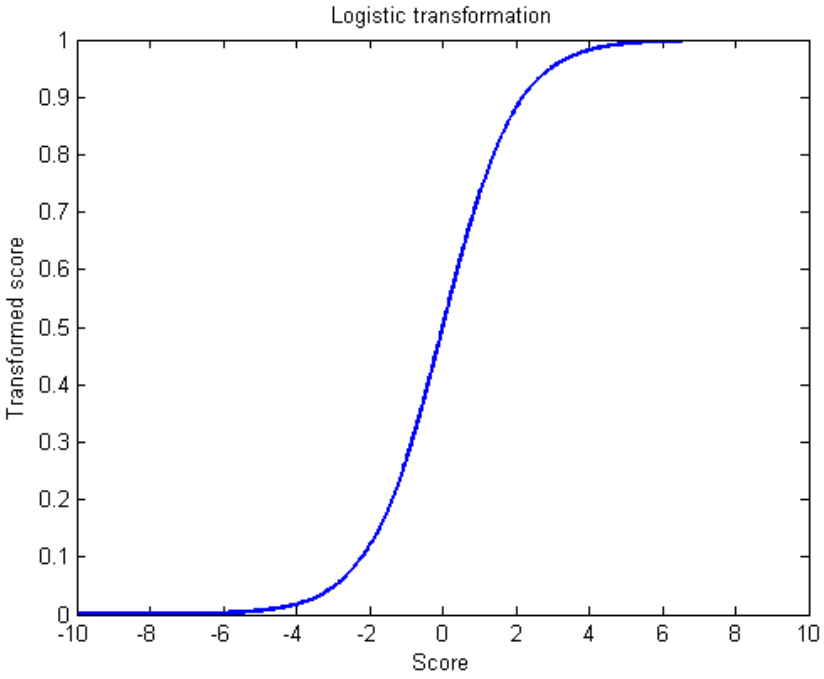

Figure 4: Logistic transformation function, in this case $a = 0$ and $b = -1$.

The formula for this transformation is:

$$f(x) = \frac{1}{1 + \exp(a + bx)}$$

The a and b in the above formula determine the horizontal translation and the steepness of the transformation. These two parameters are chosen such that the resulting logistic transformation function provides the best fit for the empirical distribution function of the factor scores. As the empirical distribution function is always monotone increasing, b will be negative for all score transformations. For those factors that are inversely related to the creditworthiness, the transformation is applied with $-a$ and $-b$.

The transformation is monotone, so the ordering of the observations according to their factor scores will remain the same. Therefore, the powerstat of a factor before transformation is the same as the powerstat of this factor after transformation.

For the transformation of the factor "total assets ratio" of the commercial banks model redevelopment, the least squares best logistic fit turned out to be $a = 26.45$ and $b = -1.06$. The graphs of the empirical distribution function of this factor and the fitted logistic function are given in Figure 5.

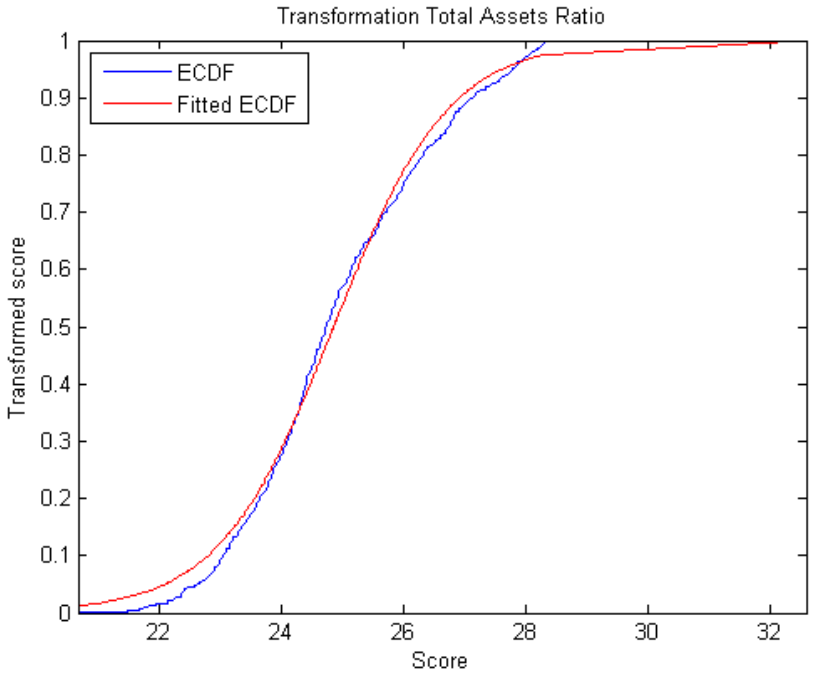

Figure 5: The empirical cumulative distribution function (ECDF) for the factor "total assets ratio" in the commercial banks model redevelopment is given, together with the fitted ECDF. This fitted ECDF is the logistic transformation for the factor scores. In this case, $a = 26.45$ and $b = -1.06$.

Through this transformation, the range of the factor scores becomes $[0,1]$. By multiplication with 10, this becomes $[0,10]$, which is the range used for all input factors of the MFA. For the four example factors (see Table 1) their transformed scores are computed and the histograms for these scores can be found in Appendix A.

2.4 Multi-Factor Analysis

During the Single Factor Analysis (SFA), the most important factors for explaining the creditworthiness of a model are identified. After transformation, these factors become the input for the Multi-Factor Analysis (MFA). During the MFA, the creditworthiness information is regressed on the factors in order to find the factor weights. The goal of the MFA is to find a model for the response variable based on the explanatory factors, while taking into account their interdependencies (Rabobank, 2010). The regression method that is used, depends on the modelling approach. If the Good-Bad approach is chosen, logistic regression is used. On the other hand, if the Shadow-Bond approach is chosen, linear regression is used.

From the SFA, a list of k factors is obtained that are the input for the MFA. From these factors, there are 2^k combinations possible for the final model. Using brute-force methods, all these combinations could be checked, but since k can be as large as 50, stepwise regression is applied. This is done for both approaches. Using this stepwise method, the model is built up step by step (Hosmer & Lemeshow, 2000). With each step, a new factor is added to the model. The first factor that is added, is the factor that – on its own – yields the model with the highest predictive power. The second factor is added by again choosing the factor that – in combination with the first factor – yields the model with the highest predictive power. This continues until no additional factor can be added that would significantly

improve the model. Also, a factor can only be added if its sign is intuitive: higher factor scores should correspond to lower PDs, so only a negative sign is allowed.

The commercial banks' portfolio does not contain many defaults, so according to 2.1.2 the Shadow-Bond modelling approach is applied for the commercial banks PD model redevelopment. On the other hand, the Poland SME portfolio contains enough defaulted observations to develop the Poland SME PD model using the Good-Bad approach. Using the development datasets for these two models, we will first discuss the MFA for the Shadow-Bond approach (2.4.1) and then proceed with the Good-Bad MFA (2.4.2).

2.4.1 Shadow-Bond MFA

For the MFA under the Shadow-Bond approach, linear regression is used. This regression method assumes that the response variable y is a linear combination of the explanatory factor scores plus a noise component. So:

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k + \epsilon$$

Here, α is an intercept, β_i denotes the factor coefficient for factor i with a factor score of x_i, $i = 1 \ldots k$. The noise is ϵ. The weights are estimated by minimizing the sum of all ϵ^2. More on this method is explained in 3.4.1.

When the Shadow-Bond approach is used, the response variable is not binary. In this case, the response variable follows from external ratings, which are mapped to a PD. The relation between the PD and the creditworthiness is assumed to be exponential (Rabobank, 2008). Then also, $\log(PD)$ and the creditworthiness are linearly related. Therefore, $y = \log(PD)$ for the linear regression.

As the Shadow-Bond approach is used for the commercial banks model redevelopment, linear regression is used during the MFA. For example, we now perform a stepwise linear regression with only the factors "total assets ratio" (x_1) and "market risk exposure" (x_2) as input. The resulting model for estimating the PD for observation j (denoted by $\widehat{PD_j}$) is then given by:

$$\widehat{PD_j} = \exp\left(-3.97 - 0.187 x_{1,j} - 0.390 x_{2,j}\right)$$

We see that "market risk exposure" is assigned a higher coefficient than "total assets ratio". If we divide each coefficient by the sum of all coefficients, we obtain the relative weights. Often, it is more insightful to discuss relative weights, instead of coefficients. The weight for the first factor, "market risk exposure", is in this case $-0.187/(-0.187 - 0.390) = 0.32$, or 32%, and for the second factor, "total assets ratio", is $-0.390/(-0.187 - 0.390) = 0.68$, or 68%.

By computing the estimates for the PD for all observations and using these as the model scores, we can compute the powerstat of the model. This is 0.746. The plot of the curves needed for the computation of the powerstat is given in Figure 6.

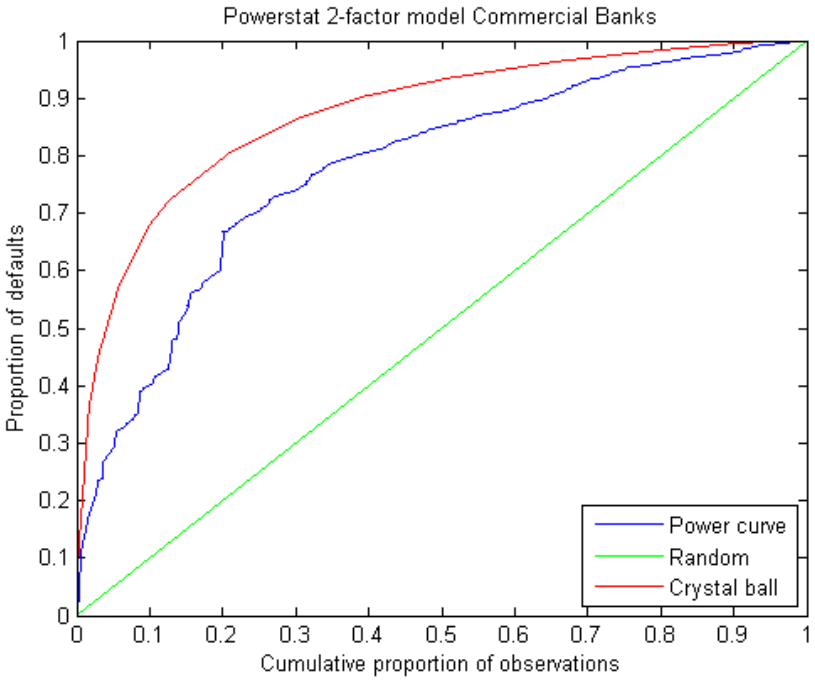

Figure 6: The curves needed for computing the powerstat of the two-factor model for commercial banks.

As the SFA already showed, the factors "total assets ratio" and "market risk exposure" have a relatively high stand-alone discriminatory power. Combining these factors in a two-factor model gives a model with an even higher powerstat. Adding more factors may further improve the model.

2.4.2 Good-Bad MFA

When the Good-Bad modelling approach is used, the response variable is binary ("default" or "non-default"). It can be shown that in that case a linear model does not provide a good fit for the data. A more detailed explanation for this is given in 3.4.2. Therefore, another (non-linear) regression model should be used. The logistic model is often used for regression on a binary response variable (Heij, De Boer, Franses, Kloek, & Van Dijk, 2004) and therefore also used in this case. The logistic model has the following expression:

$$P(y = 1|x) = \frac{1}{1 + \exp\left(-1(\alpha + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k)\right)}$$

By using maximum likelihood estimation, the α and $\beta_i, i = 1 \ldots k$ are estimated.

The Poland SME model development is based on the Good-Bad approach. Therefore, the MFA is done using stepwise logistic regression. We now suppose that we only have the factors "debt service coverage ratio" (x_1) and "history with Rabobank" (x_2) as input factors for the regression. The estimated PD for observation j is then given by:

$$\widehat{PD_j} = \frac{1}{1 + \exp\left(-1\left(-1.62 - 0.225x_{1,j} - 0.174x_{2,j}\right)\right)} = \frac{1}{1 + \exp\left(1.62 + 0.225x_{1,j} + 0.174x_{2,j}\right)}$$

Again, the relative weights can be computed from the coefficients. The weight of the first factor, "debt service coverage ratio", is $-0.225/(-0.225 - 0.174) = 0.56$, or 56%, and the

weight of the second factor, "history with Rabobank", is $-0.174/(-0.225-0.174) = 0.44$, or 44%.

By computing the estimated PD for each of the observations of the development dataset and using this as model score, we can calculate the powerstat of this two-factor model. This is 0.480. The plot of the curves needed for the computation of the powerstat is given in Figure 7.

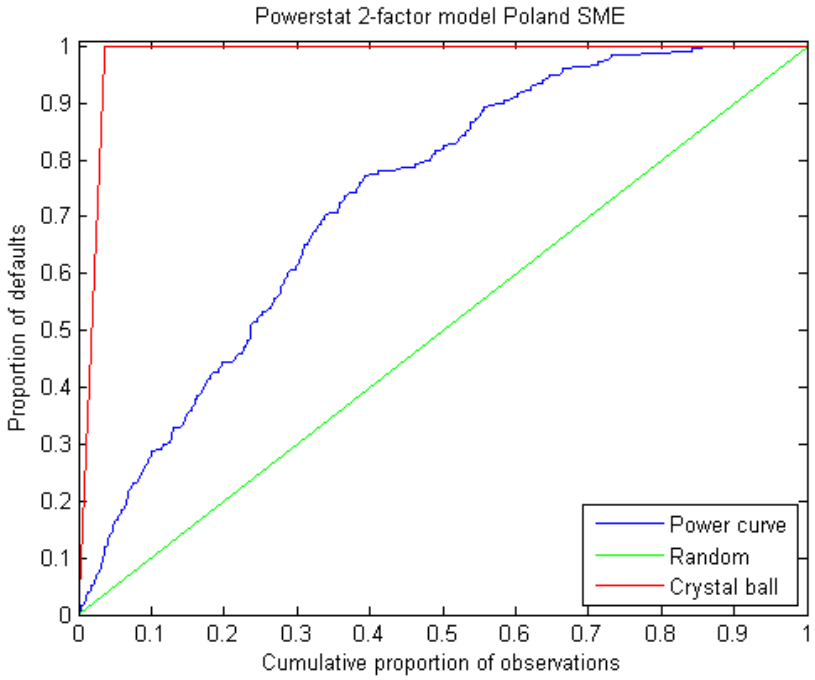

Figure 7: The curves needed for computing the powerstat of the two-factor model for Poland SME.

Again, the powerstat of the two-factor model is higher than the powerstats of the two factors separately. This suggests that adding more factors, may further increase the powerstat of the model.

2.5 Final stage

From the MFA a model is obtained for estimating the PD of an observation. During the last step of the modelling process this PD is mapped to a Rabobank Risk Rating (RRR). These RRRs are created for internal use, where R01 is the rating for counterparties with the highest creditworthiness, and R20 is the rating for counterparties with the lowest creditworthiness. The mapping from PD to RRR is done by creating *buckets* for the PD. For example, if the estimated PD for a counterparty falls between 0.4% and 0.5% – which are the boundaries of rating bucket "X" – the counterparty will be rated "X".

Furthermore, the model is tested regarding its robustness. That is, we check whether the estimated factor weights do not depend too much on the development dataset. This is done by reducing the dataset by a significant number of observations, thus creating a sample of the development dataset. The model is then computed using this sample. This is repeated for

multiple samples. If the model weights do not change too much, we can conclude that the model is robust enough (Rabobank, 2010).

The last part of the final development stage is the User Acceptance Test. During this test, future end-users of the model use the model to rate counterparties. The users give feedback on the performance of the model. If necessary, some final adjustments can be made to the model, following this feedback.

3 The ellipse method – Theory

In the previous chapter, the model development process is discussed. Now we look deeper into the modelling methodology and, specifically, the uncertainty of the *model weights*.

A rating model bases the rating of a counterparty on a certain number of characteristics that are useful in explaining its creditworthiness. These are called the *factors*, and their scores range from 0 to 10. Examples are the size of the counterparty and the quality of its management. In order to develop a rating model, the weight of each factor should be determined. The model weights represent the relative influence of each factor on the final model score.

These weights are obtained in the following way. First, some form of regression is performed on the explanatory factors and the perceived historical creditworthiness information (these could be default indicators or external PDs). The outcome of the regression is a specific coefficient for each factor. These coefficients are divided by the sum of all the coefficients to create the relative weights (as the coefficients are assumed to have the same sign and the same range).

Next, the weights are presented to a panel of experts. It often happens that the experts are not entirely in agreement with the distribution of the weights and that they suggest a shift in weight for some of the factors. It is then necessary to check whether this weight change does not deteriorate the statistical power of the model. We concentrate on the case where the weight of only *one* factor needs to be changed. For changes in multiple weights simultaneously, more research is needed.

As the weights are based on a finite dataset, they are estimates of the true weights and we can therefore imagine that there is some freedom in weight shifting. The question is, how much can these weights be changed? For determining this, we make use of two-dimensional elliptical confidence regions. Subsequently, we call this the *ellipse method*.

The two-dimensionality of the analysis can be justified as follows. Suppose the weight of factor i is changed, according to the feedback of the experts. The weights of the other factors change then as well, but proportionalities among them remain the same. In terms of coefficients this means that only the coefficient of factor i changes, the other coefficients remain unchanged. It therefore makes sense to sum all the other coefficients, thus reducing the dimensionality of the coefficients to two: the coefficient of factor i and the sum of all other coefficients.

In this chapter, the first section (3.1) discusses the general approach of the ellipse method. After that, we will discuss each of this method's steps in more detail. Section 3.2 shows how the coefficients are translated into weights and back. Section 3.3 describes how the dimensionality of the set of coefficients for the multiple factor model is reduced to two dimensions. Section 3.4 continues with finding the covariance matrix for the two specific regression methods we use in our analysis. The last two sections consider the ellipse – the confidence region for a single factor coefficient and the sum of all other coefficients. In section 3.5 the ellipse equation is derived and in section 3.6 two intervals are computed from

this ellipse. These intervals show how much the factor coefficient is allowed to change, while still being statistically supported by the data. In order to better interpret the relation between the covariance matrix and the intervals, some additional analysis on the ellipse is performed in 3.7.

3.1 General approach

The goal of the method is to find out how much the coefficient of a single factor is allowed to change without affecting the statistical power of the model. For this, we could of course use the one-dimensional confidence interval, created from the Wald statistic as described by for example Hosmer and Lemeshow (Hosmer & Lemeshow, 2000). A change in coefficient is then allowed, as long as this changed coefficient is still in the confidence interval and thus statistically supported by the data. But changing one factor coefficient may require other coefficients to change as well. To see how much change is allowed if all other coefficients should remain unchanged, we need to look at a two-dimensional region instead of the one-dimensional interval.

As we are interested in the possible change of the coefficient of a single factor i, we sum all other coefficients, thus creating a grouping coefficient. This way, the set of coefficients for the model is reduced to two dimensions. This can be done for each factor i. The grouping factor coefficient that excludes the coefficient of factor i will be referred to as the *remainder coefficient of factor i*.

Suppose the experts want to change the relative weight of factor i. As it is easier to work with coefficients instead of weights, we first compute how much the coefficient of factor i needs to change – while keeping all other coefficients fixed – in order to achieve this change in weight. How this is done is explained in section 3.2.

Next, a covariance matrix can be derived for the factor coefficients, consisting of the variance of each factor coefficient and the covariance between each pair of factor coefficients. This covariance matrix depends on the regression method used to compute the coefficients. Using this matrix, a two-by-two covariance matrix can be created for each coefficient of factor i and corresponding remainder coefficient. From this covariance matrix, an ellipse equation can be found. This ellipse is the confidence region for a predetermined confidence level and two variables: the coefficient of factor i and the remainder coefficient of factor i. See Figure 8 below for the ellipse. The coefficient of factor i is placed on the horizontal axis, the remainder coefficient of the factor i is on the vertical axis.

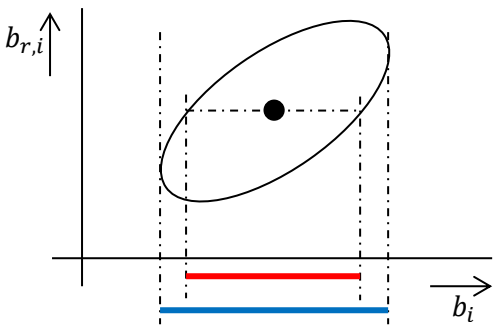

Figure 8: The confidence region of the coefficient of factor i and the remainder coefficient of factor i.

The centre of the ellipse is formed by the coefficient and the remainder coefficient of factor i, as obtained via the regression. If we want the remainder coefficient of the factor to be fixed, the coefficient of factor i can be moved over the inner interval (red), without the estimate leaving the confidence region. If we allow the remainder coefficient to change (slightly) as well, the coefficient of the factor can be moved even more, that is, over the outer interval (blue).

If we would naively construct a rectangular region with sides corresponding to the one-dimensional confidence intervals, the region would have a lower confidence level than that of the individual confidence intervals (assuming that the coefficient of factor i and remainder coefficient of factor i are not perfectly correlated). This is shown in Appendix B. The outer interval, which gives the outer bounds of the confidence region, should therefore be wider than the one-dimensional interval. The inner interval might be wider or smaller, depending on the tilt of the ellipse. So it can be noted that neither of the inner and outer intervals correspond to the one-dimensional confidence interval for the coefficient of factor i, as the effect of the other coefficient values is taken into account as well.

Considering the expert's requirements, the approach is the following. First the suggested weight change is translated back to a change in coefficient. Then, we check whether this new coefficient lies in the inner interval. If this is the case, the coefficient and corresponding weight can be changed in the model, without changing other coefficients. Since this changes the sum of the coefficients, the other weights will change slightly as well, but proportionalities among them remain the same.

On the other hand, if the new coefficient is outside of the inner interval, there are two alternative possibilities; it can either lie in the outer interval or outside it. If the new coefficient lies in the outer interval but not in the inner interval, the coefficient can be changed as proposed, but then further analysis will be needed to determine which other coefficient needs to be changed as well. This will move the estimate back into the confidence region. But if the new coefficient lies outside the blue interval, a new model analysis has to be done to determine how much the model changes and whether it is still a good model.

3.2 Weights and coefficients

In general, it is more insightful to represent a model in terms of its factor weights, instead of factor coefficients. As the weights of a good model all lie between zero and one and sum to one, it is easy to see the relative importance of an explanatory factor. However, as the weights depend on all estimated coefficients, the covariance matrix of the weights is harder to work with and far less insightful than the covariance matrix of the coefficients. Therefore, we need a method to translate a proposed change in weights into a proposed change in coefficients. Also, we should be able to represent the intervals in terms of weights, so a method to translate coefficients back into weights is needed as well.

Define b_i as the coefficient for factor i and b_i^* as the model estimate for β_i, the true underlying coefficient for factor i. So the difference between b_i and b_i^* is that b_i is the coefficient for factor i used in the model (in practice, possibly as suggested by experts), and

b_i^* is the model coefficient for factor i as calculated using a regression method. Take w_i as the weight of factor i. Then w_i is computed as follows.

$$w_i = \frac{b_i}{\sum_{j=1}^{k} b_j}$$

Here, k is the total number of explanatory factors in the model. We define the changed coefficient for factor i as $\widetilde{b_i}$ and the changed weight for factor i as $\widetilde{w_i}$.

We now consider the problem of translating a proposed change in weights into a change in coefficients. Suppose we have a model with weights w_j, $j = 1, \dots, k$ and coefficients b_j, $j = 1, \dots, k$. The experts prefer factor i to have a weight of $\widetilde{w_i}$ instead of w_i. The other weights should change while proportionally remaining constant. As we only want to change one coefficient – that is b_i – the others should remain unchanged, so $\widetilde{b_j} = b_j$ for $j \neq i$ and also $\sum_{j \neq i} \widetilde{b_j} = \sum_{j \neq i} b_j$. Then:

$$w_i = \frac{b_i}{\sum_{j=1}^{k} b_j} = \frac{b_i}{b_i + \sum_{j \neq i} b_j} \rightarrow \sum_{j \neq i} b_j = \left(\frac{1}{w_i} - 1\right) b_i$$

$$\widetilde{w_i} = \frac{\widetilde{b_i}}{\sum_{j=1}^{k} \widetilde{b_j}} = \frac{\widetilde{b_i}}{\widetilde{b_i} + \sum_{j \neq i} b_j} \rightarrow \sum_{j \neq i} b_j = \left(\frac{1}{\widetilde{w_i}} - 1\right) \widetilde{b_i}$$

From this, it follows that:

$$\left(\frac{1}{\widetilde{w_i}} - 1\right) \widetilde{b_i} = \left(\frac{1}{w_i} - 1\right) b_i$$

And subsequently:

$$\widetilde{b_i} = \frac{\left(\frac{1}{w_i} - 1\right)}{\left(\frac{1}{\widetilde{w_i}} - 1\right)} b_i \qquad (1)$$

So from the current and the proposed weight the current coefficient can be converted to the proposed coefficient, using (1).

If the intervals for a coefficient are determined, it is useful to present these in terms of weight. These can be found easily by using:

$$\widetilde{w_i} = \frac{\widetilde{b_i}}{\widetilde{b_i} + \sum_{j \neq i} b_j}$$

For $\widetilde{b_i}$ we plug in the upper and lower boundaries of the intervals.

3.3 Grouping the coefficients

If the experts are not content with one of the factor weights, it is interesting to see how much this weight can be changed so that it still remains supported by the data. In that case, only the behaviour of the coefficient in question is relevant, relative to the behaviour of the other coefficients. We can therefore group the effect of all other coefficients into one single coefficient: the remainder coefficient. This coefficient is denoted by $b_{r,i}$ with $b_{r,i} = \sum_{j \neq i} b_j$ and therefore also $w_{r,i} = \sum_{j \neq i} w_j$.

Suppose we have a covariance matrix for all k explanatory factors. The covariance matrix for the quantities b_i and $b_{r,i}$ can then also be found. Set s_i^2 as the variance of the coefficient of

factor i and $s_{i,j}$ as the covariance of the coefficients of factors i and j. Then the covariance matrix is as follows.

$$\begin{bmatrix} s_i^2 & \sum_{j \neq i} s_{i,j} \\ \sum_{j \neq i} s_{i,j} & \sum_{j \neq i} s_j^2 + \sum_{j \neq i} \sum_{k \neq i,j} s_{j,k} \end{bmatrix}$$

If the covariance matrix for all the coefficients is well defined, this matrix will be well defined as well.

3.4 Covariance matrix

In this section, we discuss the covariance matrix for the factor coefficients derived by two regression methods: the linear model with Ordinary Least Squares (OLS) estimation and the logistic model with Maximum Likelihood Estimation (MLE). The second method is preferred if the response variable is binary, so if it can only have two outcomes (for example, for the Good-Bad modelling approach) (Heij, De Boer, Franses, Kloek, & Van Dijk, 2004). In other cases, the linear model is used (for example, for the Shadow-Bond modelling approach).

3.4.1 Linear model

This model is of the form $y = x\beta + \epsilon$ where y is the dependent variable (for the Shadow-Bond method, $y = \log(PD)$), x is a row vector of explanatory factors, β is a column vector of coefficients for the factors, and ϵ is an error term. To find the OLS estimator of β, we need to use the available dataset of observations. The β that minimizes the sum of all ϵ^2's is the OLS estimator for β. Using the data of the N observations, we get the following matrix notation of the model:

$$Y = X\beta + \epsilon$$

Here, Y is the $N \times 1$ vector of dependent variables, β is a $k \times 1$ vector of coefficients, X is a $N \times k$ matrix with each row containing the factor scores of one observation, and ϵ is a $N \times 1$ vector of errors with mean 0. Minimizing the sum of ϵ^2's is equivalent to minimizing $\epsilon^T\epsilon$, which again corresponds to minimizing $(Y - X\beta)^T(Y - X\beta)$. From this, the OLS estimator b^* for β follows, which is $b^* = (X^TX)^{-1}X^TY$. The derivation is shown in Appendix C.

It can be shown that b^* is an unbiased estimator for β:

$$E(b^*) = E((X^TX)^{-1}X^TY) = (X^TX)^{-1}X^TE(Y) = (X^TX)^{-1}X^TX\beta = \beta$$

Furthermore, we assume that the variance of ϵ is given by $\sigma^2 I$, where I is the identity matrix and σ is fixed. This means that the ϵ_i's have the same variance and are uncorrelated. We can then find the covariance matrix of the estimator b^*. First note:

$$b^* - \beta = (X^TX)^{-1}X^T(X\beta + \epsilon) - \beta = (X^TX)^{-1}X^T\epsilon$$

Then also:

$$(b^* - \beta)(b^* - \beta)^T = (X^TX)^{-1}X^T\epsilon\epsilon^TX(X^TX)^{-1}$$

And therefore:

$$Var(b^*) = (X^TX)^{-1}X^TE(\epsilon\epsilon^T)X(X^TX)^{-1} = \sigma^2(X^TX)^{-1}X^TX(X^TX)^{-1} = \sigma^2(X^TX)^{-1}$$

This is the covariance matrix for the OLS estimator b^*.

In order to say more on the uncertainty of b^*, the covariance matrix is indispensable. But for this matrix, σ^2 is needed. As the errors ϵ are unobservable but necessary in the estimation of

σ^2, they need to be substituted. Since b^* is an estimator for β, we find $e = Y - Xb^*$ as a natural substitute for ϵ. It can then be shown that an unbiased estimator for σ^2 is given by

$$s^2 = \frac{e^T e}{N - k}$$

Also, under the assumption that Y is N-variate normal, it follows that $(N - k)s^2/\sigma^2$ is distributed as $\chi^2(N - k)$: it follows a chi-squared distribution with $N - k$ degrees of freedom. Furthermore, b^* and s are distributed independently (Theil, 1971). This is required in section 3.5.

3.4.2 Logistic model

For some applications the response variable is binary. This is for example the case under the Good-Bad modelling approach: an observation is either in default or not in default. Under the linear model, the conditional mean of this binary response variable (y) is expressed as a linear equation in x: $E(y|x) = x\beta$, where β denotes the factor coefficients. But now that y is binary, this conditional mean is restricted to lie between 0 and 1. Figure 9 shows how a linear and a non-linear model are fitted to a binary dependent variable.

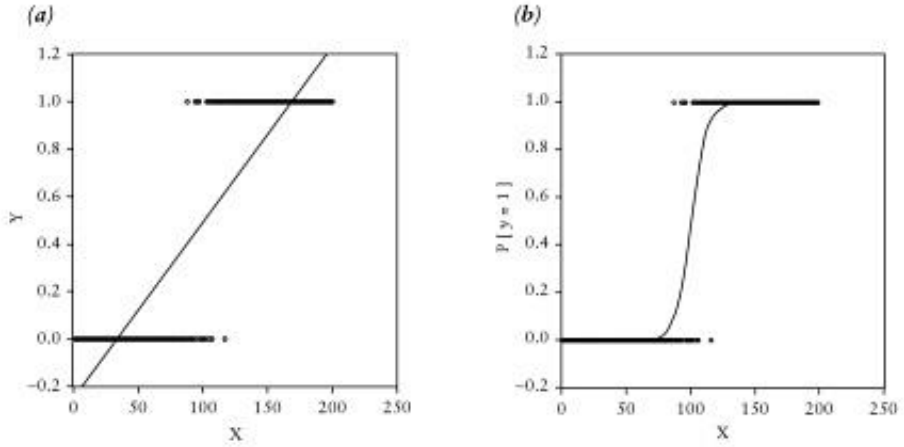

Figure 9: Binary dependent variable with a linear model (a) and with a non-linear model (b) for a single explanatory variable (x). Figure from Heij et al. (Heij, De Boer, Franses, Kloek, & Van Dijk, 2004).

From the left side of Figure 9 we can see that the linear model would not provide a good fit for the binary response. Also, the linear model may yield theoretically inadmissible values: $E(y|x) < 0$ or $E(y|x) > 1$. The model on the right side of Figure 9 seems to provide a better fit for the binary response. The S-shaped curve shows that this is not a linear model. Indeed, this *logistic model* has the following expression:

$$\ln\left(\frac{E(y|x)}{1 - E(y|x)}\right) = x\beta$$

This can be rewritten to:

$$E(y|x) = \frac{1}{1 + e^{-x\beta}}$$

Since y is either 0 or 1, $E(y|x)$ has a value between 0 and 1. So, it can also be interpreted as $P(y = 1|x)$. Therefore, F can in turn be interpreted as a cumulative distribution function, if F is defined as:

$$F(x\beta) := \frac{1}{1 + e^{-x\beta}} \tag{2}$$

If we would again use OLS estimation to estimate β, the error term for observation i is defined as $\epsilon_i = y_i - F(x_i\beta)$. Note that x_i is a row vector of the factor scores for observation i. Since the response variable is binary, either $\epsilon_i = 1 - F(x_i\beta)$ with probability $F(x_i\beta)$ or $\epsilon_i = -F(x_i\beta)$ with probability $1 - F(x_i\beta)$. It follows that:

$$Var(\epsilon_i) = F(x_i\beta)\big(1 - F(x_i\beta)\big)$$

It can thus be seen that the variance of ϵ_i depends on x_i and the ϵ_i's are therefore heteroscedastic, i.e. $Var(\epsilon_i)$ is not identical for all i. But in order to use the conventional formulas for the standard error in the OLS results in this chapter, homoscedasticity is required. That is, $Var(\epsilon_i)$ should be identical for all i (see Verbeek (Verbeek, 2004)). Since this is not the case, OLS cannot be used and we will therefore use another estimation method: Maximum Likelihood Estimation (MLE).

When MLE is used, β is estimated from the available observation data. This means that the probability of observing the available observations is maximized. For observation i, the probability of $y_i = 1$ is $F(x_i\beta)$. Similarly, the probability of $y_i = 0$ is $1 - F(x_i\beta)$. Denoting $p_i = F(x_i\beta)$ we find the likelihood function $L(\beta)$of β under $x_1, x_2, \dots, x_N$:

$$L(\beta) = \prod_{i;y_i=1} p_i \prod_{i;y_i=0} (1 - p_i)$$

The log-likelihood is then:

$$\log L(\beta) = \sum_{i;y_i=1} \log(p_i) + \sum_{i;y_i=0} \log(1 - p_i) = \sum_{i=1}^{N} y_i \log(p_i) + \sum_{i=1}^{N} (1 - y_i) \log(1 - p_i)$$

Maximizing this would yield the same result as maximizing the likelihood function. The β that maximizes $\log L(\beta)$ is the maximum likelihood estimator b^*. It can be found by solving $\frac{\partial}{\partial\beta} \log L(\beta) = 0$.

$$\begin{aligned} \frac{\partial}{\partial\beta} \log L(\beta) &= \sum_{i=1}^{N} \frac{y_i}{p_i} \frac{\partial p_i}{\partial\beta} + \sum_{i=1}^{N} \frac{1 - y_i}{1 - p_i} \frac{\partial(1 - p_i)}{\partial\beta} = \sum_{i=1}^{N} \frac{y_i}{p_i} f_i x_i - \sum_{i=1}^{N} \frac{1 - y_i}{1 - p_i} f_i x_i \\ &= \sum_{i=1}^{N} \frac{y_i - p_i}{p_i(1 - p_i)} f_i x_i = \sum_{i=1}^{N} (y_i - p_i) x_i = 0 \end{aligned} \tag{3}$$

Here, f_i denotes $f(x_i\beta)$, the density function of F as in (2). The next-to-last equality (3) is explained by the following:

$$f_i = f(x_i\beta) = \frac{e^{-x_i\beta}}{(1 + e^{-x_i\beta})^2} = \frac{1}{1 + e^{-x_i\beta}} \left(1 - \frac{1}{1 + e^{-x_i\beta}}\right) = F(x_i\beta)\big(1 - F(x_i\beta)\big) = p_i(1 - p_i)$$

By solving the first order conditions from (3), the b^* is found.

In order to use the ellipse method on the obtained b^*, the covariance matrix is needed. Heij et al. (Heij, De Boer, Franses, Kloek, & Van Dijk, 2004) showed that the inverse of the information matrix I_N evaluated at the b^* found by logistic regression is a good approximation for the covariance matrix of b^*. This I_N is given by:

$$I_N = -E\left[\frac{\partial^2 \log L(\beta)}{\partial\beta\partial\beta^T}\right]$$

So we have to find the second derivative to β of $\log L$.

$$\frac{\partial^2 \log L(\beta)}{\partial\beta\partial\beta^T} = \frac{\partial \sum_{i=1}^{N} (y_i - p_i) x_i}{\partial\beta^T} = -\sum_{i=1}^{N} x_i \frac{\partial p_i}{\partial\beta^T} = -\sum_{i=1}^{N} f_i x_i^T x_i = -\sum_{i=1}^{N} p_i(1 - p_i) x_i^T x_i$$

The covariance matrix of b^* is therefore given by:

$$Var(b^*) = \left(\sum_{i=1}^{N} F(x_i b^*)\,(1 - F(x_i b^*)\,) x_i^T x_i \right)^{-1}$$

3.5 Constructing the ellipse

In the previous sections we reduced a model with multiple factors to a two-factor model and created the covariance matrix for the coefficient of factor i and the remainder coefficient of factor i (the sum of all other coefficients). For these two coefficients, we can now construct a two-dimensional confidence region, using the covariance matrix. This region will be ellipse-shaped, due to the different variances of the two coefficients. So if we find the equation for this ellipse, the confidence region will be uniquely defined. First, we derive the ellipse equation for the coefficients following from the linear regression model. We then extend this to the coefficients of the logistic regression model.

3.5.1 Linear model

In this subsection we discuss the derivation of the ellipse equation for the coefficients calculated by the linear model. Suppose we fix a factor i for whose coefficient and remainder coefficient this ellipse will be created. For this subsection, we will follow the explanation of Theil (Theil, 1971).

Remember that the covariance matrix of b^* from the linear regression model was given by $\sigma^2(X^T X)^{-1}$, where X denotes the $N \times k$ data matrix. Using the method described in section 3.3, we can create the two-by-two covariance matrix for the coefficient of factor i and the remainder coefficient of factor i. Denote this matrix by $\sigma^2 G$.

As G is symmetric and positive definite, there exist a non-singular matrix P such that $G^{-1} = P^T P$ (for example, its Cholesky decomposition (Trefethen & Bau, 1997)). Introduce z_1, z_2 in the following way:

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = P \begin{bmatrix} b_i^* - \beta_i \\ b_{r,i}^* - \beta_{r,i} \end{bmatrix}$$

Remember that b_i^* is the model estimate for β_i, the underlying "real" coefficient. If again Y is assumed to be N-variate normal, it follows that $[z_1, z_2]^T$ is a random normal vector with zero mean and the following covariance matrix:

$$Var([z_1, z_2]^T) = P\,Var\left(\left[b_i^* - \beta_i, b_{r,i}^* - \beta_{r,i}\right]^T\right) P^T = \sigma^2 P G P^T = \sigma^2 P P^{-1} (P^T)^{-1} P^T = \sigma^2 I$$

It follows that both z_1/σ and z_2/σ are independent and standard normally distributed. Therefore, $z_1^2 + z_2^2$ is distributed as $\sigma^2 \chi^2(2)$. Also, remember that $(N-k)s^2$ is distributed as $\sigma^2 \chi^2(N-k)$ and that b^* and s are distributed independently. As it is known (see for example Zijp (Zijp, 1974)), if U_1 and U_2 are independent and both chi-squared distributed with d_1 and d_2 degrees of freedom respectively, then

$$V = \frac{U_1/d_1}{U_2/d_2} \sim F(d_1, d_2)$$

So in that case, V follows an F-distribution with parameters d_1 and d_2. This can be applied to our current situation as well.

$$\frac{z_1^2 + z_2^2}{2} = \frac{1}{2}\left(\left[b_i^* - \beta_i, b_{r,i}^* - \beta_{r,i} \right] P^T P \begin{bmatrix} b_i^* - \beta_i \\ b_{r,i}^* - \beta_{r,i} \end{bmatrix} \right) = \frac{1}{2}\left(\left[b_i^* - \beta_i, b_{r,i}^* - \beta_{r,i} \right] G^{-1} \begin{bmatrix} b_i^* - \beta_i \\ b_{r,i}^* - \beta_{r,i} \end{bmatrix} \right) \sim \sigma^2 \chi^2(2)/2 \quad (4)$$

Also,

$$s^2 \sim \sigma^2 \chi^2(N-k)/(N-k) \quad (5)$$

Combining (4) and (5), we can therefore say that

$$\frac{1}{2s^2}\left(\left[b_i^* - \beta_i, b_{r,i}^* - \beta_{r,i} \right] G^{-1} \begin{bmatrix} b_i^* - \beta_i \\ b_{r,i}^* - \beta_{r,i} \end{bmatrix} \right) \sim F(2, N-k)$$

Using this, we can finally create a confidence region for β_i and $\beta_{r,i}$; the coefficient and remainder coefficient of factor i.

$$P\left(\frac{g_{11}}{2s^2}(b_i^* - \beta_i)^2 + \frac{g_{12}}{s^2}(b_i^* - \beta_i)(b_{r,i}^* - \beta_{r,i}) + \frac{g_{22}}{2s^2}(b_{r,i}^* - \beta_{r,i})^2 \leq F_{1-\alpha} \right) = \alpha$$

Here, α is the chosen confidence level, for example 95%, and $F_{1-\alpha}$ is the corresponding critical value. Furthermore, g_{ij} denotes the (i,j)th element of G^{-1}. It is clear from the expression within the probability brackets, that the confidence region for the two coefficients is an ellipse around $\left(b_i^*, b_{r,i}^*\right)$.

3.5.2 Logistic model

As many models are developed using the logistic regression model, it is useful to extend the ellipse method to coefficients derived with this form of regression as well. The method for deriving the ellipse equation for coefficients of the logistic model is similar to the method for deriving the ellipse equation for coefficients of the linear model. An important difference is that the covariance matrix for the coefficients of the logistic model is an approximation, whereas the covariance matrix for the coefficients of the linear model is exact.

From subsection 3.4.2 we found (the approximation for) the covariance matrix for the coefficients b^* derived with the logistic regression model.

$$Var(b^*) = \left(\sum_{i=1}^{N} F(x_i b^*)\,(1 - F(x_i b^*))\, x_i^T x_i \right)^{-1} \quad (6)$$

It follows directly that this can only be an approximation of $Var(b^*)$, as the right side of (6) also depends on b^*. But since it has been shown that this approximation converges in probability to the covariance matrix (Heij, De Boer, Franses, Kloek, & Van Dijk, 2004) and since in practice the dataset is large enough, we will use (6) as the covariance matrix.

Again using the method as described in 3.3 the covariance matrix for the coefficient for factor i and the remainder coefficient can be created from the covariance matrix for all coefficients. This two-by-two covariance matrix is referred to as H. As H is symmetric and positive definite, there is a non-singular matrix Q such that $H^{-1} = Q^T Q$ (for example, its Cholesky decomposition (Trefethen & Bau, 1997)). Then, define w_1, w_2 such that:

$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = Q \begin{bmatrix} b_i^* - \beta_i \\ b_{r,i}^* - \beta_{r,i} \end{bmatrix}$$

The MLE estimate for β follows an asymptotic normal distribution (Heij, De Boer, Franses, Kloek, & Van Dijk, 2004) so we can say that b^* is approximately normally distributed with mean β and covariance matrix as given above. Here, we will assume that b^* is indeed

normally distributed. Then $[w_1, w_2]^T$ is also normally distributed with mean zero and covariance matrix as given by:

$$Var([w_1, w_2]^T) = Q\ Var\left(\left[b_i^* - \beta_i, b_{r,i}^* - \beta_{r,i}\right]^T\right) Q^T = QHQ^T = QQ^{-1}(Q^T)^{-1}Q^T = I$$

Therefore, w_1 and w_2 are independent standardized normal variables. So $w_1^2 + w_2^2 \sim \chi^2(2)$. This can be written out as:

$$w_1^2 + w_2^2 = \left[b_i^* - \beta_i, b_{r,i}^* - \beta_{r,i}\right] Q^T Q \begin{bmatrix} b_i^* - \beta_i \\ b_{r,i}^* - \beta_{r,i} \end{bmatrix} = \left[b_i^* - \beta_i, b_{r,i}^* - \beta_{r,i}\right] H^{-1} \begin{bmatrix} b_i^* - \beta_i \\ b_{r,i}^* - \beta_{r,i} \end{bmatrix}$$

Using this distribution, a confidence region can be created.

$$P\left(h_{11}(b_i^* - \beta_i)^2 + 2h_{12}(b_i^* - \beta_i)\left(b_{r,i}^* - \beta_{r,i}\right) + h_{22}\left(b_{r,i}^* - \beta_{r,i}\right)^2 \leq \chi_{1-\alpha}^2\right) = \alpha$$

Again, α is the confidence level and $\chi_{1-\alpha}^2$ denotes the critical value. Also, $h_{i,j}$ is the (i,j)th element of H^{-1}. The confidence region is an ellipse around $\left(b_i^*, b_{r,i}^*\right)$.

3.6 Deriving the intervals

From the ellipse equations as found in section 3.5 we can now derive the intervals that illustrate the confidence we have in the estimates for the factor coefficients. For each coefficient, two intervals can be created. One of these intervals is nested in the other. This *inner* interval is the interval over which the coefficient in question can be moved without having to change any other coefficient. The other interval – the *outer* interval – shows the values the coefficient can have, that are still supported by the data. Changing the coefficient to a value that is outside the inner interval, but still inside the outer interval, requires a modification of at least one other coefficient as well.

Remember that the regression estimates for β_i and $\beta_{r,i}$ are given by b_i^* and $b_{r,i}^*$. We found that the ellipse equation for coefficients derived with the linear OLS regression is given by:

$$\frac{g_{11}}{2s^2}(b_i^* - \beta_i)^2 + \frac{g_{12}}{s^2}(b_i^* - \beta_i)\left(b_{r,i}^* - \beta_{r,i}\right) + \frac{g_{22}}{2s^2}\left(b_{r,i}^* - \beta_{r,i}\right)^2 = F_{1-\alpha}$$

For coefficients derived with the logistic MLE regression the ellipse equation is:

$$h_{11}(b_i^* - \beta_i)^2 + 2h_{12}(b_i^* - \beta_i)\left(b_{r,i}^* - \beta_{r,i}\right) + h_{22}\left(b_{r,i}^* - \beta_{r,i}\right)^2 = \chi_{1-\alpha}^2$$

As these two equations both generate ellipses, we can generalize them to one ellipse equation and find the intervals from that equation. Also, as we are now trying to find coefficients (denoted by b) instead of the *real* underlying parameter (denoted by β), we write b instead of β. This general equation is then:

$$A(b_i^* - b_i)^2 + B(b_i^* - b_i)\left(b_{r,i}^* - b_{r,i}\right) + C\left(b_{r,i}^* - b_{r,i}\right)^2 = 1 \quad (7)$$

If the linear regression method is used, $A = g_{11}/(2s^2F_{1-\alpha})$, $B = g_{12}/(s^2F_{1-\alpha})$, and $C = g_{22}/(2s^2F_{1-\alpha})$. If, on the other hand, the logistic regression method is used, $A = h_{11}/\chi_{1-\alpha}^2$, $B = 2h_{12}/\chi_{1-\alpha}^2$, and $C = h_{22}/\chi_{1-\alpha}^2$. Throughout this section, ellipse equation (7) will be used.

Now that the ellipse equation is known, the intervals for b_i can be calculated. The first interval can be found by fixing the remainder coefficient of the factor i. The second interval is computed by splitting the ellipse into two parts and finding the boundaries for b_i for both parts.

3.6.1 Inner interval

The inner interval is the smallest interval. If the experts suggest a changed coefficient that still lies within this interval, the change can be executed directly without having to change other

coefficients. Suppose the experts suggest changing the coefficient for factor i. Not changing any other coefficient implies that the remainder coefficient for factor i – which is the sum of all other coefficients – remains the same. So by fixing $b_{r,i} = b_{r,i}^*$ the bounds for b_i around b_i^* can be found. This is illustrated in Figure 10.

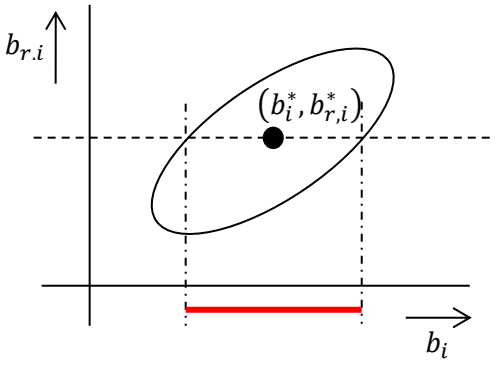

Figure 10: The red interval is the inner interval. It is computed by fixing $b_{r,i} = b_{r,i}^*$.

If we fix $b_{r,i} = b_{r,i}^*$, the ellipse equation will simplify to:

$$A(b_i^* - b_i)^2 + B(b_i^* - b_i)\left(b_{r,i}^* - b_{r,i}^*\right) + C\left(b_{r,i}^* - b_{r,i}^*\right)^2 = A(b_i^* - b_i)^2 = 1$$

Solving this quadratic equation for b_i yields two solutions; those are the boundaries of the inner interval. Thus, the inner interval is given by:

$$\left[b_i^* - \frac{1}{\sqrt{A}}, b_i^* + \frac{1}{\sqrt{A}}\right]$$

The width of this inner interval is therefore $2/\sqrt{A}$.

3.6.2 Outer interval

The outer interval is slightly wider than the inner interval. If the proposed changed factor coefficient lies outside the inner interval but within the outer interval, the change can be approved, but other factor coefficients may need to be changed as well.

Computing the outer interval is done by splitting the ellipse equation into two parts. This split is shown in Figure 11, together with the outer interval.

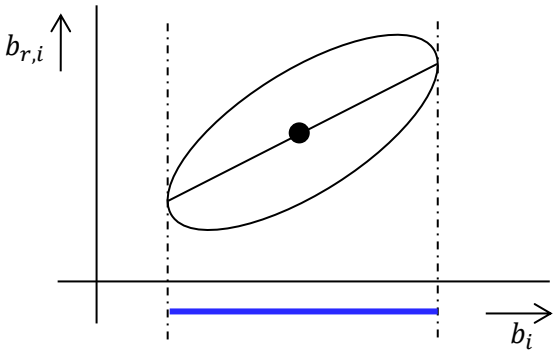

Figure 11: The blue interval is the outer interval. The line through the ellipse splits the ellipse in two parts.

The ellipse is split in such a way, that every value for b_i corresponds to one value on each arc. (If there is a value for b_i that corresponds to two or more values on the same arc, the splitting

is not done correctly.) Therefore, the outer b_i values of each arc are also the outer b_i values of the ellipse. These values are found by determining the domain of the expressions for the arcs. In order to find the expressions for the arcs, (7) has to be rewritten in such a way that $b_{r,i}$ is expressed in terms of b_i. This gives:

$$b_{r,i} = -\frac{B}{2C}(b_i^* - b_i) \pm \frac{\sqrt{(b_i^* - b_i)^2(B^2 - 4AC) + 4C}}{2C} + b_{r,i}^*$$

Note that these are actually two different expressions, due to the $\pm$-sign. In order for these expressions to be real, it should hold that $(b_i^* - b_i)^2(B^2 - 4AC) + 4C \geq 0$. Keeping in mind that $B^2 - 4AC < 0$ (as A, B, and C follow from an equation for an ellipse) the interval for b_i follows. The outer interval is therefore given by:

$$\left[b_i^* - \sqrt{\frac{-4C}{B^2 - 4AC}}, b_i^* + \sqrt{\frac{-4C}{B^2 - 4AC}}\right] = \left[b_i^* - \frac{1}{\sqrt{A - \frac{B^2}{4C}}}, b_i^* + \frac{1}{\sqrt{A - \frac{B^2}{4C}}}\right]$$

The width of this interval is $2/\sqrt{A - B^2/4C}$.

In order to check whether the inner interval is indeed smaller than the outer interval, we have to verify if indeed $2/\sqrt{A} \leq 2/\sqrt{A - B^2/4C}$. Note that if $B = 0$, the two intervals would be identical. If, on the other hand, $B \neq 0$ then $B^2/4C > 0$ (since C is strictly positive if (7) indeed defines an ellipse). It then follows directly that $2/\sqrt{A} \leq 2/\sqrt{A - B^2/4C}$. As both intervals are centred around b_i^*, the inner interval always lies within the bounds of the outer interval.

3.7 Analysis of the ellipse and the intervals

In order to better interpret the relation between the covariance matrix and the intervals, we now perform some additional analysis.

The two intervals derived in 3.6 follow from the confidence region for the coefficient of factor i and the remainder coefficient of factor i. The shape of this region is an ellipse. As can be seen from Figure 8, the two intervals would be identical, if the major and minor axes of the ellipses would be parallel to the axes of b_i and $b_{r,i}$. If this is not the case (and assuming that b_i and $b_{r,i}$ have a different variance), the ellipse is somewhat tilted. To get more insight in the ellipse and the intervals that follow from it, we need to know the cause of the tilt of the ellipse. The expression for the tilt (θ) is given by:

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{B}{A - C}\right) \tag{8}$$

The derivation for this expression is given in Appendix D.

In order to find an interpretation for the ratio $B/(A - C)$, we need to go back to the covariance matrix for the factor coefficients. Recall that from this matrix, a two-by-two covariance matrix could be created for each set of estimated coefficients b_i^* and $b_{r,i}^*$. It consists of variances and covariances:

$$\begin{bmatrix} Var(b_i) & Cov(b_i, b_{r,i}) \\ Cov(b_i, b_{r,i}) & Var(b_{r,i}) \end{bmatrix}$$

Its inverse, which is used to create A, B, and C for the ellipse equation, is given by:

$$\frac{1}{Var(b_i) * Var(b_{r,i}) - Cov(b_i, b_{r,i})^2} * \begin{bmatrix} Var(b_{r,i}) & -Cov(b_i, b_{r,i}) \\ -Cov(b_i, b_{r,i}) & Var(b_i) \end{bmatrix}$$

We can now use these values to find the ratio $B/(A-C)$. Remember that A, B, and C were computed differently for the coefficients derived by linear regression than for those derived by logistic regression. For linear regression we have:

$$\frac{B}{A-C} = \frac{\frac{g_{12}}{s^2 F_{1-\alpha}}}{\frac{g_{11}}{2s^2 F_{1-\alpha}} - \frac{g_{22}}{2s^2 F_{1-\alpha}}} = \frac{2g_{12}}{g_{11} - g_{22}} = \frac{2\frac{g_{12}}{\sigma^2}}{\frac{g_{11}}{\sigma^2} - \frac{g_{22}}{\sigma^2}}$$

As the g_{jk}/σ^2 with $j, k = 1, 2$ are the entries of the inverse of the two-by-two covariance matrix, we obtain:

$$\frac{B}{A-C} = \frac{-2Cov(b_i, b_{r,i})}{Var(b_{r,i}) - Var(b_i)}$$

The same result follows for the coefficients derived by logistic regression:

$$\frac{B}{A-C} = \frac{\frac{2h_{12}}{\chi^2_{1-\alpha}}}{\frac{h_{11}}{\chi^2_{1-\alpha}} - \frac{h_{22}}{\chi^2_{1-\alpha}}} = \frac{2h_{12}}{h_{11} - h_{22}} = \frac{-2Cov(b_i, b_{r,i})}{Var(b_{r,i}) - Var(b_i)}$$

Plugging this into (8) gives:

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{-2Cov(b_i, b_{r,i})}{Var(b_{r,i}) - Var(b_i)}\right) \quad (9)$$

It can therefore be seen that the larger the covariance between the two coefficients, the bigger the angle over which the ellipse is tilted. Also, the smaller the difference in variances, the bigger the angle. In the extreme case, if the variances of b_i and $b_{r,i}$ are equal, then the ellipse will be tilted over an angle of $\pi/4$ radians, or 45°.

If the angle of the tilt of the ellipse is large, this does not necessarily mean that the difference in width between the inner and outer interval is large as well. This difference strongly depends on the shape of the ellipse as well. This is illustrated in Figure 12 and Figure 13, where both ellipses have the same tilt but a different shape.

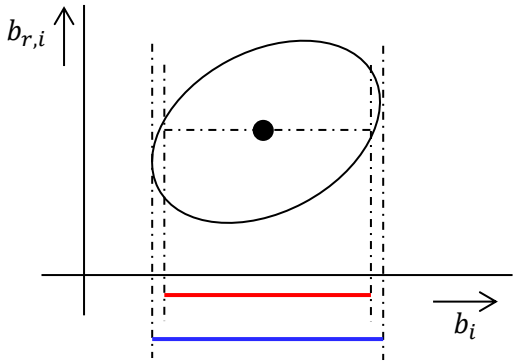

Figure 12: The confidence region of the coefficient of factor i and the remainder coefficient of factor i. The difference between the width of the two intervals is small.

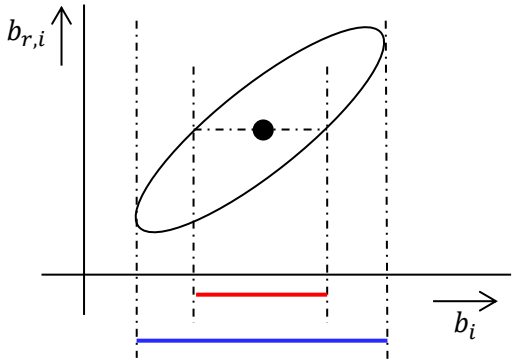

Figure 13: The confidence region of the coefficient of factor i and the remainder coefficient of factor i. The difference between the width of the two intervals is large.

In order to find out what determines the difference in width of the inner and outer interval, we divide the width of the outer interval by the width of the inner interval. The bigger the difference between the two intervals, the larger this ratio will be. Recall that the width of the inner interval is given by $2/\sqrt{A}$ and that of the outer interval by $2/\sqrt{A - B^2/4C}$. The ratio of these widths is therefore given by:

$$\frac{\frac{2}{\sqrt{A - \frac{B^2}{4C}}}}{\frac{2}{\sqrt{A}}} = \sqrt{\frac{A}{A - \frac{B^2}{4C}}} = \frac{1}{\sqrt{1 - \frac{B^2}{4AC}}}$$

It follows that the higher the value of the ratio $B^2/4AC$, the larger the difference in intervals. By again using the expressions for A, B, and C, we obtain for the coefficients found by linear regression:

$$\frac{B^2}{4AC} = \frac{\left(\frac{g_{12}}{s^2 F_{1-\alpha}}\right)^2}{4 * \frac{g_{11}}{2s^2 F_{1-\alpha}} * \frac{g_{22}}{2s^2 F_{1-\alpha}}} = \frac{\left(\frac{g_{12}}{\sigma^2}\right)^2}{\frac{g_{11}}{\sigma^2} * \frac{g_{22}}{\sigma^2}} = \frac{Cov^2(b_i, b_{r,i})}{Var(b_{r,i}) * Var(b_i)} \tag{10}$$

For logistic regression, the same result follows:

$$\frac{B^2}{4AC} = \frac{\left(\frac{2h_{12}}{\chi^2_{1-\alpha}}\right)^2}{4 * \frac{h_{11}}{\chi^2_{1-\alpha}} * \frac{h_{22}}{\chi^2_{1-\alpha}}} = \frac{h_{12}^2}{h_{11} * h_{22}} = \frac{Cov^2(b_i, b_{r,i})}{Var(b_{r,i}) * Var(b_i)} \tag{11}$$

The right side of (10) and (11) looks familiar; in fact, it is the square of the correlation ρ_i:

$$\rho_i := \frac{Cov(b_i, b_{r,i})}{\sqrt{Var(b_{r,i}) * Var(b_i)}}$$

Note that this ρ_i is the correlation between the coefficient of factor i and the remainder coefficient of factor i (the sum of all other coefficients). It follows that the ratio of the widths for the intervals of factor i can be written as $1/\sqrt{1 - \rho_i^2}$. So the more b_i and $b_{r,i}$ are correlated, the bigger the difference between the width of the inner interval and the width of the outer interval.

4 The ellipse method – Application

In the previous chapter we developed a method to create intervals for the estimated factor coefficients of the rating model. These intervals can be used to provide more insight in how much the development data supports the possible changes to the coefficients as suggested by experts. The method that creates these intervals is called *the ellipse method*. This name refers to the shape of the confidence region that is used for creating the intervals.

The ellipse method works as follows. First, a specific factor i is chosen for whose coefficient b_i the intervals are required. All the other coefficients are summed and this coefficient is called the *remainder coefficient of the factor* i, denoted by $b_{r,i}$. As the covariance matrix for the coefficient values is known, the two-by-two covariance matrix for b_i and $b_{r,i}$ can be created. Using this two-by-two matrix, an ellipse-shaped confidence region can be made for b_i and $b_{r,i}$. From this confidence region, two intervals follow. These are illustrated in Figure 14.

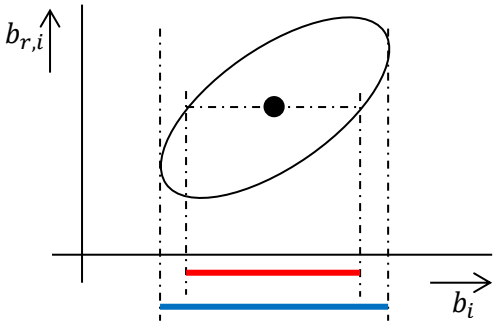

Figure 14: The confidence region of the coefficient of factor i and the remainder coefficient of factor i.

In Figure 14, the inner interval (red) is the interval over which b_i can be changed – possibly following experts' recommendations – without having to change any other coefficients. The inner interval contains all the values to which b_i may be changed, that are statistically supported by the data. If b_i is changed to a value that is outside the inner interval, but inside the outer (blue) interval, at least one other coefficient has to change as well. For changing more than one coefficient simultaneously, similar but more elaborate analysis is needed, which is outside the scope of this thesis.

In this chapter, we will provide numerical background for the theory of the previous chapter. First, we will verify by means of Monte Carlo simulation that the created ellipses are indeed confidence regions. This is done in 4.1. From these ellipses, the intervals are created. To see if these intervals are useful in practice and to give some illustration of the method, we use the ellipse method on "real" development datasets: datasets that are used for the development of rating models for Rabobank. This is described in 4.2. Also, in order to give an idea of how the ellipse method works on highly correlated factors, an example of a two-factor model is created in 4.3, consisting of two factors with a relatively high correlation. For Rabobank, the Matlab implementation of the ellipse method is explained in Appendix E.

4.1 Checking the confidence region

The intervals that are created using the ellipse method rely on the ellipse being the confidence region for b_i and $b_{r,i}$. In addition to the mathematical derivation in 3.5, we also

show numerically that this ellipse is indeed the required confidence region. This is done to confirm the correctness of the derivation and to show that sufficiently good results are achieved by using an approximation for the covariance matrix of the coefficients that follow from logistic regression.

The coefficients $b_i, i = 1 \dots k$ are actually estimates for the "real" underlying parameters $\beta_i, i = 1 \dots k$. Therefore, an interpretation for the α-confidence region around $(b_i, b_{r,i})$ is that $(\beta_i, \beta_{r,i})$ lies within this region with a confidence level of α. We use this interpretation for the numerical tests. For this, a *Monte Carlo method* is used. For this type of method, many random simulations are performed. In this case the following steps were taken:

1. A β is fixed and based on this β a "universe" of many random observations is generated.
2. A random sample is taken from all the observations in the universe.
3. On this sample, b is estimated.
4. Using the covariance matrix of b, an ellipse is created around $(b_i, b_{r,i})$ (i is predetermined).
5. We check if $(\beta_i, \beta_{r,i})$ lies within this ellipse.
6. The steps 2 to 5 are repeated many times.

It is important to note that the number of observations in the universe is far more than the number of observations per sample.

Figure 15 gives an illustration of this Monte Carlo method.

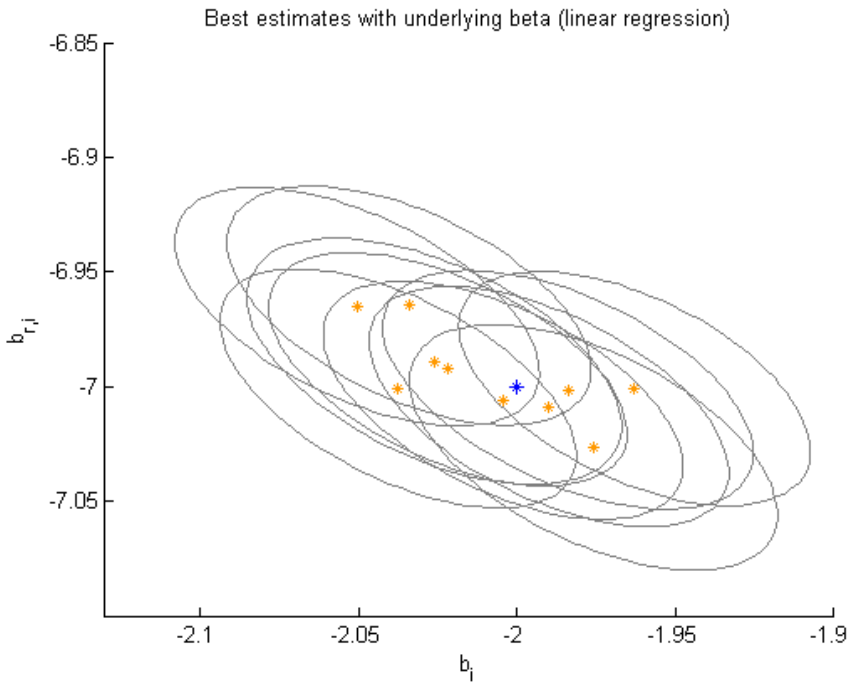

Figure 15: Illustration of the Monte Carlo method that is used to check if the ellipse is indeed a confidence region for b_i and $b_{r,i}$ (estimated with linear regression). The blue dot is β, and the orange dots are the estimates b for β. The corresponding ellipse is drawn around each estimate.

In Figure 15 the blue dot represents β. For this plot, only 10 Monte Carlo iterations are performed. Each iteration provided an estimate b (orange dots) and the corresponding ellipse around b. It can be seen that in all cases β lies within the ellipse.

Depending on the modelling approach used for the model (re)development, either linear or logistic regression is used. The covariance matrix for the coefficient estimates following from linear regression is defined differently from the covariance matrix for the coefficient estimates following logistic regression. We will therefore perform the Monte Carlo simulations on both regression methods separately.

4.1.1 Linear regression

First, we discuss the results for the linear regression method. This regression method is used if the response variable is not binary, which is the case if the Shadow-Bond modelling approach is applicable. Similar to Figure 15, the estimates b and underlying parameter β are plotted in Figure 16. The ellipses are not plotted for clarity reasons (there are 10000 Monte Carlo simulations). The confidence level was set to be 95%.

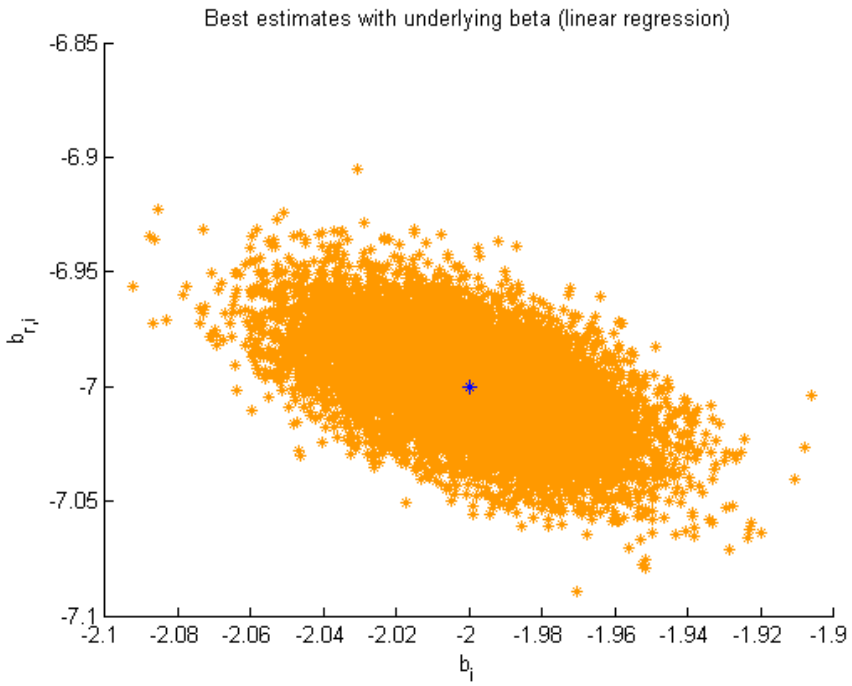

Figure 16: The blue dot is β, and the orange dots are the estimates b, found by linear regression.

It followed that for 10000 Monte Carlo simulations of 2000 observations each, the percentage of ellipses around b for which β was inside the ellipse was 95.04%. We can see these simulations as two-sided binomial tests with the outcome "β in ellipse" as success. If we set $P(\beta\ in\ ellipse) = 95\%$ as null hypothesis and $P(\beta\ in\ ellipse) \neq 95\%$) as the alternative hypothesis, the outcome 95.04% out of 10000 binomial tests has a p-value of 86%. The p-value is the probability of obtaining a result that is at least as extreme as the one observed, assuming the null hypothesis holds. We can therefore safely assume that the ellipse indeed defines a confidence region with confidence level 95%.

4.1.2 Logistic regression

Now, the same is done for the other regression method: the logistic regression. This method is used when the Good-Bad modelling approach is applicable. This is the case if the response variable is binary, for example if its answers can only be "default" or "non-default". Again, we use 10000 Monte Carlo simulations and plot the estimates b and the β. See Figure 17.

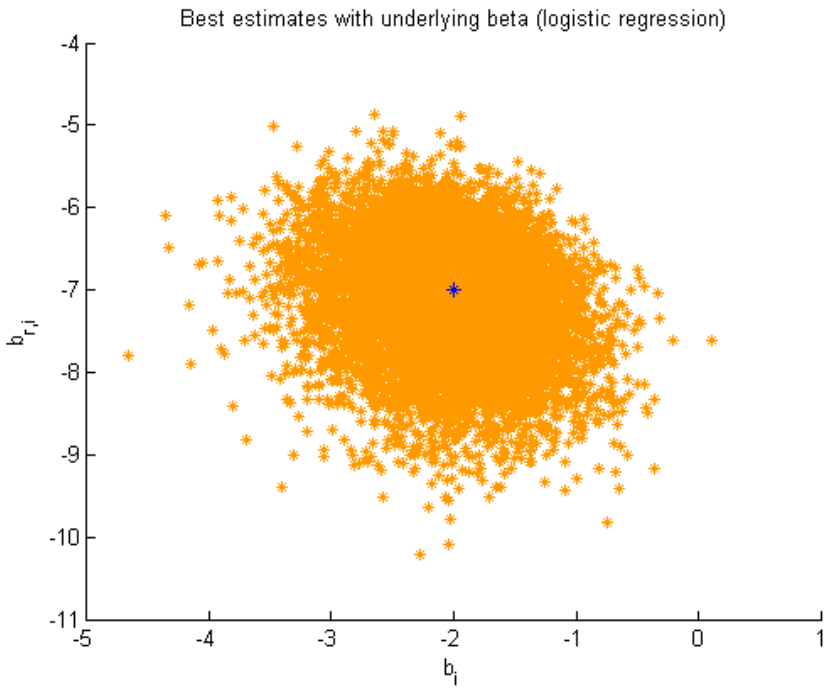

Figure 17: The blue dot is β, and the orange dots are the estimates b, found by logistic regression.

For 10000 Monte Carlo simulations of 2000 observations each, 95.25% of the ellipses contained β. By again regarding these simulations as binomial tests and taking null hypothesis $P(\beta\ in\ ellipse) = 95\%$ and alternative hypothesis $P(\beta\ in\ ellipse) \neq 95\%$, we can find the p-value for the outcome of 95.25% for 10000 tests. This p-value is 26%. This is high enough to safely assume that the ellipse indeed defines a 95% confidence region.

By comparing the scales of Figure 16 and Figure 17, something else can be noticed. Apparently, the confidence regions for the coefficients obtained by logistic regression is much larger than the regions for the coefficients obtained by linear regression. This follows from the difference in dependent variable for linear regression and logistic regression. As the dependent variable is binary for the logistic regression, it gives less information regarding the creditworthiness than the dependent variable for linear regression, which has more answering categories. Therefore, the variance of the factor coefficients is higher for those derived by logistic regression. So it can also be expected that the intervals around coefficients derived from logistic regression are wider than those for coefficients derived from linear regression. The results of the next section will confirm this.

4.2 Interval examples

If a statistically developed rating model is presented to the experts, they may suggest modifications of the factor coefficients. To see how much the modifications are supported by

the data, two intervals are created per coefficient. The inner interval shows the values that the coefficient is allowed have, under the condition that the other coefficients remain the same. The outer interval contains all the values the coefficient might be changed to that are still statistically supported by the dataset.

To provide some more illustration, the intervals are created for models based on the datasets for the commercial banks and Poland SME PD model (re)developments.

4.2.1 Intervals for the commercial banks model redevelopment

For the commercial banks model redevelopment the Shadow-Bond approach is used, including the linear regression. Through stepwise regression, ten factors are selected and estimates for their coefficients are found. For each coefficient, we can then create an ellipse. These are depicted in Figure 18.

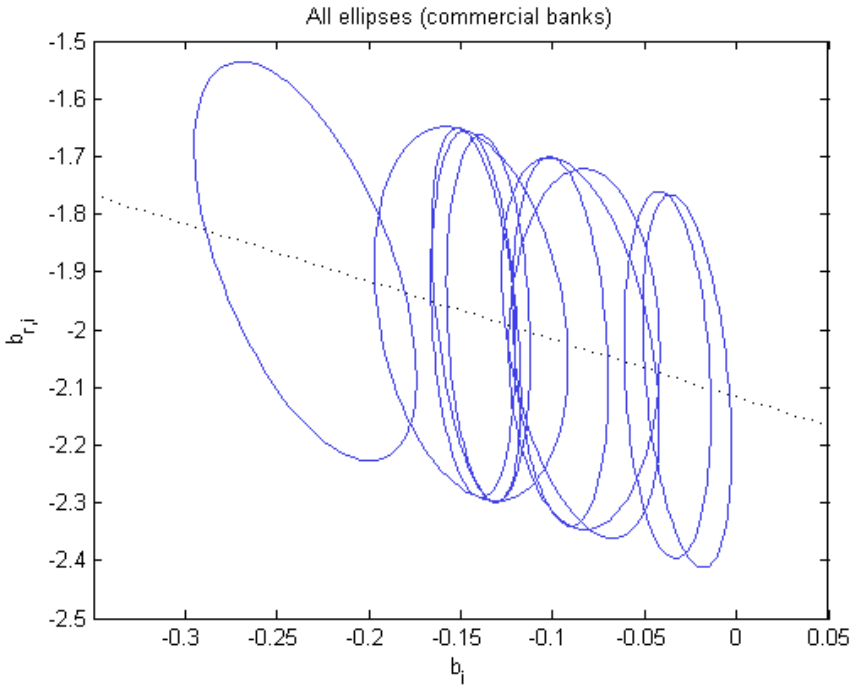

Figure 18: The ellipses for the coefficients of the ten factors that are included in the model for commercial banks.

The values on the axis of Figure 18 do not denote b_i and $b_{r,i}$ for fixed i, but the i depends on the ellipse we look at. Actually, this plot is a combination of the ten plots of individual ellipses. The dotted line goes through the centres of all ellipses. It makes sense that this is a line, since $b_i^* + b_{r,i}^*$ – the sum of all estimated coefficients – is the same for each i.

When we look at the minor axes of the ellipses in Figure 18, we can see that five of them are relatively wide, whereas the other five are relatively thin. This difference can be related to the types of factors considered. In general, the variance of the scores for the financial factors is higher than that for the qualitative factors. This makes it easier to identify the different groups of creditworthiness. Therefore, the coefficients for the financial factors can be estimated more precisely, leading to a lower variance for their factor coefficients. This is

reflected in the shape of the ellipses for the coefficients of the financial factors: they are "thinner" than those for the coefficients of the qualitative factors, since their coefficient value is already quite precise and can therefore not change too much.

It can be seen from Figure 18 that the ellipses that form the confidence regions for each b_i and $b_{r,i}$ are not very tilted. That is, except for one ellipse, their major and minor axes are almost parallel to the b_i and $b_{r,i}$ axes. The reason for this can be found in the number of factors and the weak correlation between the factor coefficients. In this case, $Var(b_{r,i})$ is much larger than $Var(b_i)$, which reduces the angle of the tilt of the ellipse (see (9) in 3.7).

Since it is more insightful to discuss factors' contributions in terms of weights, instead of coefficients, the intervals for the coefficients are translated to intervals in terms of weights (see 3.2). For the ten factors used in the commercial banks model, the intervals (in weighs) are given in Table 4.

		Inner interval		Outer interval	
Factor number	Weight	Lower weight	Upper weight	Lower weight	Upper weight
1	20.8%	17.2%	24.1%	16.3%	24.9%
2	12.8%	8.7%	16.6%	8.5%	16.7%
3	12.7%	10.9%	14.4%	10.9%	14.5%
4	12.6%	10.8%	14.2%	10.6%	14.4%
5	12.0%	10.2%	13.7%	10.2%	13.7%
6	8.5%	6.4%	10.5%	6.4%	10.6%
7	7.5%	4.2%	10.6%	3.9%	10.9%
8	7.4%	3.8%	10.6%	3.8%	10.6%
9	3.3%	1.3%	5.3%	1.3%	5.3%
10	2.4%	0.4%	4.3%	0.2%	4.4%

Table 4: The inner and outer intervals for the factor weights of the commercial banks model.

It can be seen that the inner and outer intervals for each factor weight do not differ much. As a matter of fact, for the factors 5, 8, and 9 the inner and outer intervals are (almost) identical. From 3.7 we can therefore conclude that the correlation between each coefficient and its remainder coefficient is small.

4.2.2 Intervals for the Poland SME model development

For the Poland SME model development the Good-Bad approach is used. This includes performing a logistic regression. Eight factors are included through stepwise regression and for each of these factors the coefficient in the model is estimated. The ellipses for these factor coefficients are depicted in Figure 19.

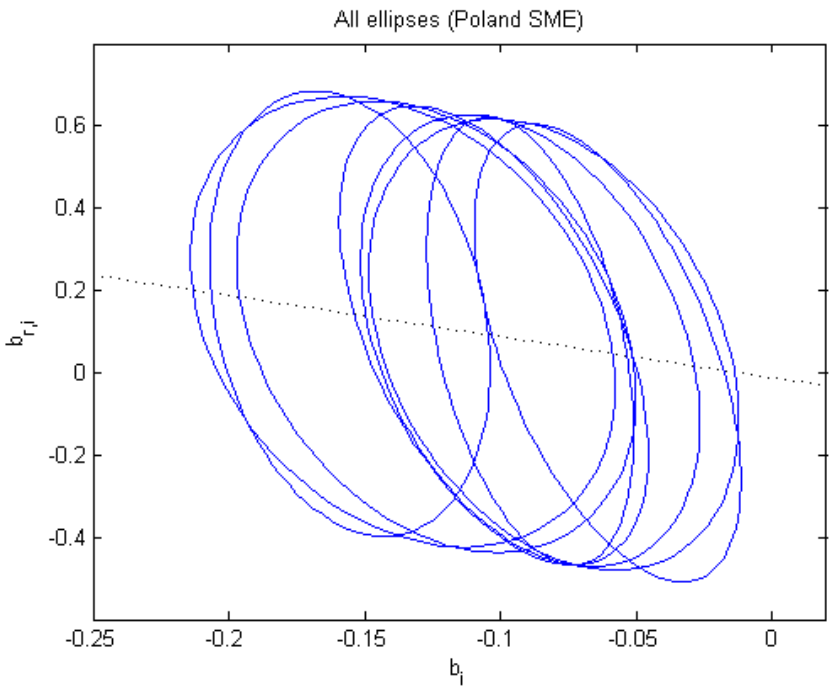

Figure 19: The ellipses for the coefficients of the eight factors that are included in the model for Poland SME.

Again, the dotted line goes through the centres of all ellipses. From these ellipses, the inner and outer intervals for each factor coefficient can be created, and these can be translated in terms of weights. The inner and outer intervals for the weights of the eight factors used in the Poland SME model are given in Table 5.

		Inner interval		Outer interval	
Factor number	Weight	Lower weight	Upper weight	Lower weight	Upper weight
1	18.6%	13.4%	23.2%	13.2%	23.3%
2	16.2%	8.0%	23.1%	7.6%	23.4%
3	14.8%	7.0%	21.4%	6.6%	21.7%
4	12.3%	6.8%	17.1%	5.8%	17.8%
5	10.6%	4.0%	16.5%	3.4%	16.9%
6	10.6%	6.8%	14.2%	6.4%	14.5%
7	9.6%	2.1%	16.1%	1.6%	16.4%
8	7.2%	2.5%	11.5%	1.5%	12.4%

Table 5: The inner and outer intervals for the factor weights of the Poland SME model.

From Table 5 we see that the intervals for the factor weights are relatively wide. This means that there is quite some freedom in modifying the factor weights, statistically supported by the dataset. An explanation for these intervals being wider than those in Table 4 can be found in considering the type of dependent variable used for logistic regression (as used for the Poland SME data). This variable is binary and thus contains less information than the dependent variable for linear regression (as used for the commercial banks data), which has more than two answering categories. As there is less information available on the creditworthiness of the observations, there is more uncertainty in the estimated weights for

the explanatory factors. Therefore, the intervals for coefficients estimated by logistic regression are in general larger than those for coefficients estimated by linear regression.

Similar to the intervals for the factor weights of the commercial banks model, the differences between the inner and outer intervals for the factors weights of the Poland SME model are small. This can also be seen from Figure 19. The major and minor axes of most ellipses are not rotated very much, compared to the b_i and $b_{r,i}$ axes.

4.3 High correlation example

From the examples in 4.2 we see that in practice the differences between the inner and the outer intervals are small. This follows from the low correlation between the coefficients of the factors. A cause for this low correlation between the factor coefficients is the low correlation between the factors. Highly correlated factors are in general not incorporated in the model. If two factors have a high correlation, either one is omitted from the model or the two are combined into one new factor (Rabobank, 2010). Also, the large number of factors in the model reduces the correlation between factor coefficients and their remainder coefficients.

So in order to investigate the effects of a strong correlation, we synthetically create a model, consisting of only two factors, preferably with a relatively high correlation. Therefore, two factors from the commercial banks model are chosen – one financial and one qualitative – that have a correlation of 0.21. This is relatively high for two factors of different types that are allowed to both be included in the commercial banks rating model.

Now, a model is built consisting of only these two factors using linear regression. The model coefficients have a high (as in: close to -1) correlation of -0.85. Note that this is the correlation between two factor *coefficients*, whereas we saw that 0.21 denotes the correlation between the factors itself. This high correlation of -0.85 follows from the (relatively) high correlation between the factors, and also from the fact that these two are the only two factors in the model. The ellipse for the two factor coefficients is given in Figure 20.

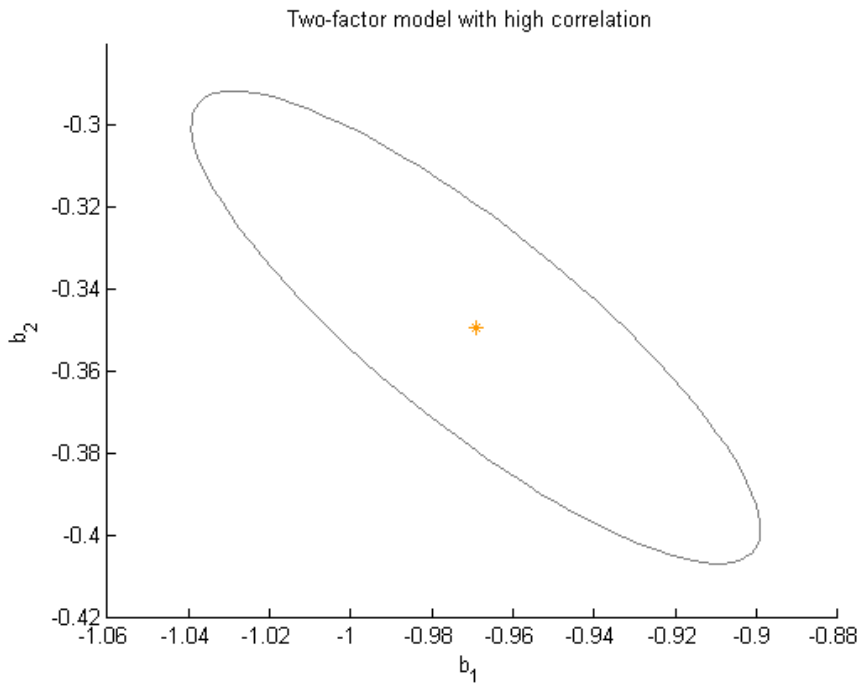

Figure 20: The confidence region for the two factor coefficients (the only two factors in the model).

It can be seen that the ellipse in Figure 20 is very tilted. This is caused by the small difference in the variance compared to the high covariance of the two factor coefficients. Also, the high correlation between the coefficients causes a big difference in width of the inner and the outer intervals for the coefficients. These are given – in terms of weights – in Table 6.

		Inner interval		Outer interval	
Factor	Weight	Lower weight	Upper weight	Lower weight	Upper weight
1	73.5%	72.8%	74.2%	72.0%	74.8%
2	26.5%	24.8%	28.1%	23.1%	29.6%

Table 6: The inner and outer intervals for the factor weights of the two-factor model.

In this case, it can be seen that for both factors, the width of the outer interval is almost twice that of the inner interval. Remember from 3.7 that the ratio of the widths of the outer interval and the inner interval can be computed directly from ρ_i, the correlation between the factor coefficients b_i and $b_{r,i}$ (here, b_1 and b_2). The expression for this ratio is given by $1/\sqrt{1-\rho_i^2}$. Plugging in $\rho_i = -0.85$ yields a ratio of 1.90. This confirms that the width of the outer interval is almost double that of the inner interval. Note that the widths of the intervals in terms of coefficients (instead of weights) will have a ratio of exactly 1.90.

We see that if we create a model consisting of only two factors that are relatively strongly correlated, their coefficients have a very high correlation. This high correlation between the factor coefficients causes a big difference in the widths for the outer and the inner intervals.

5 Powerstat & divergence – Theory

In Chapter 3 we developed a method to quantify the uncertainty of the factor weights that are calculated using regression. When all factor weights are approved by experts, we have obtained a model that can give a score to an observation, based on qualitative and financial information (factors). The next step in the rating model development is to check some of this model's characteristics. In this chapter, we would like to quantify the uncertainty for one of the statistics used for these checks: the measure for the *discriminatory power* of the model.

After developing a rating model, it is sensible to compare the model predictions with the observed reality – within the development sample itself – and to check how well the model can discriminate between observations with different levels of creditworthiness. This model characteristic is called the discriminatory power. Also, after using the model for some time, this test can be performed on a set of new observations, thus testing if the model is still up-to-date.

The performance measure for the discriminatory power that is most often used by Rabobank, is the *powerstat* (a precise definition is given in 5.1.1). We already introduced the powerstat in Chapter 2. However, in this chapter we will discuss the powerstat from a more theoretical point of view.

The powerstat of a model has a value between -1 and 1. The higher the powerstat, the better the discriminatory power of the model. But from a statistical point of view, a single powerstat value contains little information. It is therefore useful to also provide a confidence interval for the powerstat, indicating the interval in which the model's powerstat should lie with a certain level of confidence.

By modifying the model (for example, by adding an extra factor) a higher powerstat can be obtained. One may think: "the higher the powerstat, the higher the discriminatory power of the model". But to be able to say that the modification leads to a *significant* change in the powerstat, the new powerstat should be outside the confidence interval of the old powerstat. So only when the new powerstat is higher than the upper bound of the confidence interval of the old powerstat, we can conclude that the new model has a significantly higher powerstat. This way, the confidence interval of the powerstat can be used to check whether the effort of modifying the model is paid back in terms of a higher discriminatory power.

It is not obvious how to calculate the confidence interval for the powerstat, especially from the way it is defined (algorithmically). Instead, we use an alternative measure for the discriminatory power: the *divergence*. Numerical results suggest that under normality assumptions these two measures are equivalent, but no formal proof is given yet. If these two measures are indeed equivalent, this would enable the extension of some desirable properties of the divergence to the powerstat – including the creation of a confidence interval.

In this chapter, we first give an introduction of the powerstat and the divergence (section 5.1). We proceed with the proof of the equivalence of the two measures in section 5.2. After this, the confidence interval is created in section 5.3. Also, in this section another application of

the equivalence of the two measures is given: a new interpretation of the powerstat. Throughout this chapter, it is assumed that an observation is either "good" (non-default) or "bad" (default), thus following the Good-Bad modelling approach. The other approach – the Shadow-Bond approach, where observations are divided over more than two groups according to their external rating – is not considered. In section 5.4 the main reasons for this are given.

5.1 Introduction to powerstat & divergence

In this section, we introduce the two performance measures powerstat and divergence. They will be the key players in this chapter, so a proper introduction is required. We start with the measure that is most often used in practice: the powerstat.

5.1.1 Powerstat

The powerstat is a performance measure for the discriminatory power and has much in common with the well-known Gini coefficient (Gini, 1912). The Gini coefficient is usually used for measuring the income dispersion in a country. The coefficient is close to 0 if all inhabitants have almost the same income. On the other hand, the Gini coefficient reaches its maximal value 1 if one person gains all the income of the whole country. As many countries aim for income equality, a high Gini coefficient is interpreted as a negative result.

Similar to the Gini coefficient, the absolute value of the powerstat ranges between 0 and 1. But whereas one in general tries to minimize the Gini coefficient, the powerstat is maximized. This is because the powerstat is used to measure the discriminatory power of a rating model. Therefore: the higher the powerstat, the better the model.

We now focus on the computation of the powerstat. Let S_G denote the model score of a good observation, S_B the model score of a bad observation, and S_T the model score of an arbitrary observation (good or bad). Similarly, let F_G denote the distribution function of S_G, F_B that of S_B, and F_T that of S_T. For a given score s the following expressions should hold.

$$P(S_G \leq s) = F_G(s) \tag{12}$$

$$P(S_B \leq s) = F_B(s) \tag{13}$$

If we assume that the number of observations is infinite, the Probability of Default (PD) is known, as this equals the observed default frequency. We therefore have:

$$P(S_T \leq s) = F_T(s) = (1 - PD)F_G(s) + PD\ F_B(s) \tag{14}$$

Note that (14) is a linear combination of (12) and (13).

In order to be able to link the powerstat with the divergence, some assumptions are needed in this chapter. We assume that both the model scores of the good and bad observations are normally distributed with mean and standard deviation μ_G, σ_G, and μ_B, σ_B, respectively. So it follows that $F_G(s) = N(s; \mu_G, \sigma_G)$ and $F_B(s) = N(s; \mu_B, \sigma_B)$. Also, it is safe to assume that the model scores of the good observations and those of the bad observations are independently distributed. The probability of having an observation (good or bad) with model score lower than (or equal to) s follows then from (14):

$$P(S_T \leq s) = F_T(s) = (1 - PD)\ N(s; \mu_G, \sigma_G) + PD\ N(s; \mu_B, \sigma_B)$$

For computing the powerstat, the area under the power curve is needed. The power curve is an example of a *probability-probability plot* (P-P plot): one cumulative distribution function is plotted against another cumulative distribution function. For the power curve, these are F_B and F_T – the cumulative distribution functions of the model scores of the bad observations and the model scores of all observations (good and bad), respectively. So for a certain score s the probability of a bad observation having a model score worse than (or equal to) s is plotted against the probability of an arbitrary observation having a model score worse than (or equal to) s. This is illustrated in Figure 21.

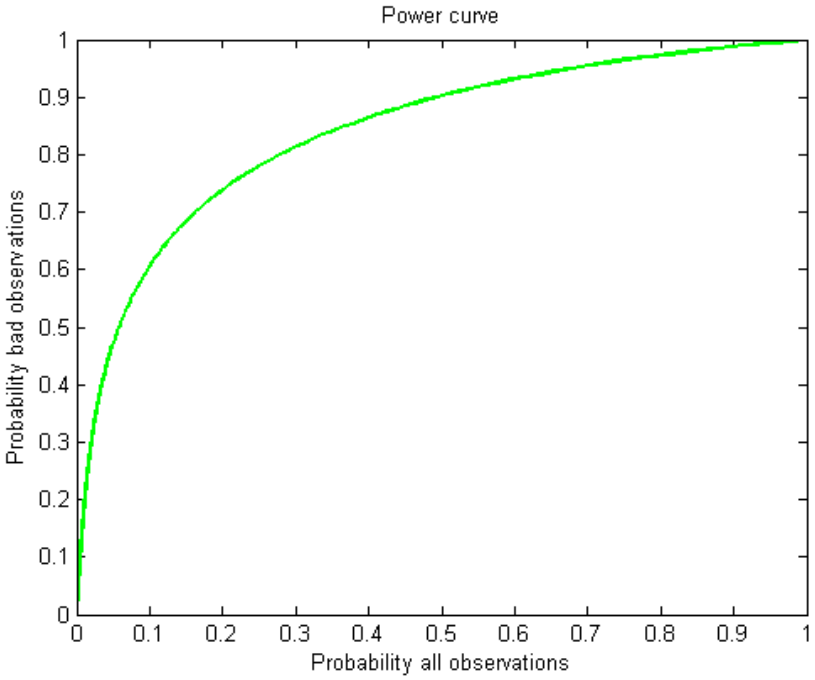

Figure 21: The power curve is obtained by plotting the cumulative distribution function of model scores of bad observations (F_B) against the cumulative distribution function of model scores of all observations (F_T).

For the computation of the powerstat, two other curves are needed in addition to the power curve. These are the curves corresponding to the *crystal ball model* and the *random model*. If we were able to exactly predict which counterparties survive and which would default – as if we had a crystal ball – we would have the best model possible. For example, if there are 100 defaults in 1000 observations, the 100 worst model scores would be given to the 100 defaulting counterparties. The curve would first be a straight line with slope $1000/100 = 10$, and then continue as a horizontal line.

The random model is the model that has no discriminatory power at all. The model scores are distributed independently of the creditworthiness of the counterparties i.e. $F_B = F_G = F_T$. The curve of this model is therefore a straight line with slope $F_B/F_T = 1$.

The power curve and the curves of the crystal ball model and random model are given in Figure 22.

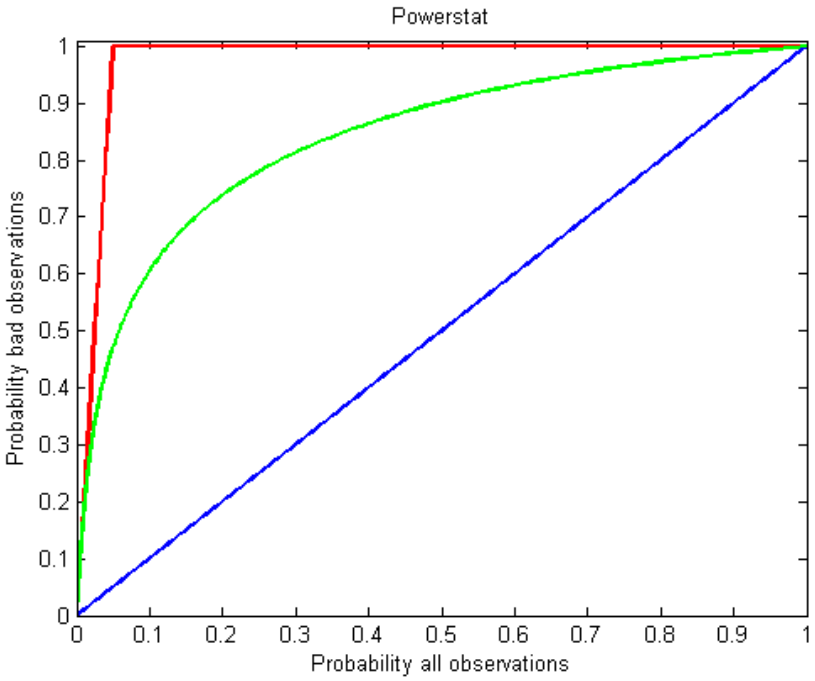

Figure 22: The power curve is given in green. The red and blue curves represent the crystal ball model and the random model, respectively.

The powerstat (denoted by PS) is computed as a ratio of two areas. The area between the power curve and the curve for the random model is divided by the area between the curve for the crystal ball model and the curve for the random model. So the closer the power curve lies to the curve for the crystal ball model, the higher the powerstat. This can be expressed mathematically by:

$$PS = \frac{A_{PC} - A_{RM}}{A_{CB} - A_{RM}} = \frac{2A_{PC} - 1}{1 - PD} \tag{15}$$

Here, A_{PC} is the area under the power curve, A_{CB} is the area under the curve for the crystal ball model, and A_{RM} represents the area under the curve for the random model, which is always ½. The area under the curve for the crystal ball model A_{CB} is equal to $1 - PD/2$.

In order to compute the area under the power curve, we first need its mathematical expression. Since the power curve is the relation between the distribution functions for the model scores of the bad observations and all observations, this is given by:

$$p_B = pc(p_T) = F_B(F_T^{-1}(p_T))$$

The area under the power curve is then:

$$A_{PC} = \int_0^1 pc(p_T)dp_T \tag{16}$$

Then, (16) can be plugged into (15) to compute the powerstat.

5.1.2 Divergence

The discriminatory power of a model can also be measured by the so-called *divergence* (Siddiqi, 2006). It is based on the distance between the two mean model scores, relative to the standard deviations of both distributions. This means that the smaller the overlap between the two distributions, the higher the divergence. The divergence can only yield a

sensible result if the model scores for the good and bad observations are normally distributed. Under the same normality assumptions as given in subsection 5.1.1, the divergence is computed by:

$$D = \frac{2(\mu_B - \mu_G)^2}{\sigma_B^2 + \sigma_G^2}$$

From this expression it follows that if the distributions for the model scores of the good and bad observations are further apart, the divergence is higher. Also, if the standard deviations are smaller, the divergence increases as well. This is illustrated in Figure 23: the larger the difference in the means or the smaller the variances, the smaller the overlap in the distributions of the model scores for the good and bad observations.

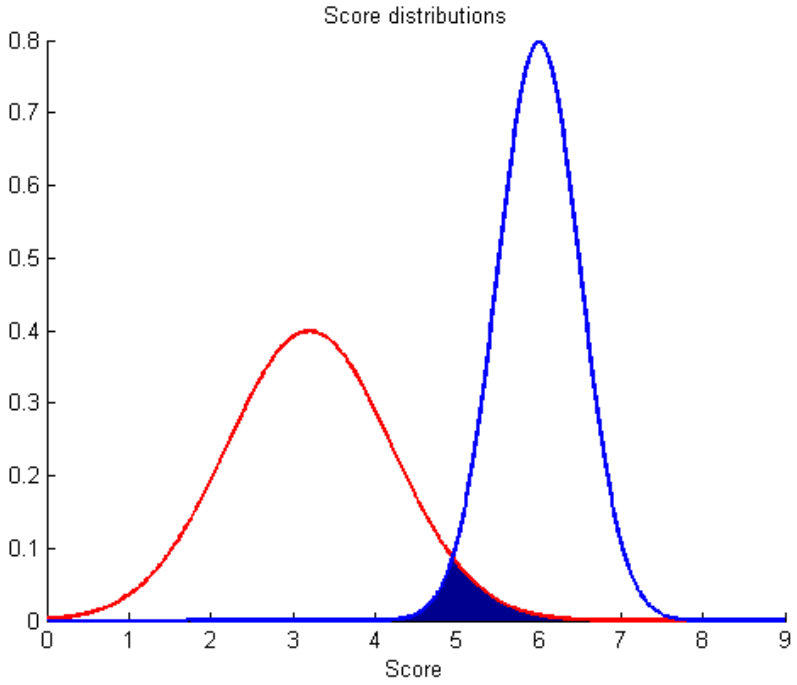

Figure 23: The model score distributions for bad (left) and good (right) observations.

It is clear that the divergence is very easy to compute, and this property will be used later in this chapter.

5.2 Equivalence of the performance measures

The powerstat and divergence are both performance measures for the discriminatory power of a rating model. From numerical tests, we noticed that for a model with a higher powerstat, the divergence is also higher. Conversely, a model with a lower powerstat also has a lower divergence. These tests therefore suggest that there might be an equivalence relation between the powerstat and the divergence. In this section, we would like to answer the main question "Can the powerstat be mapped one-to-one to the divergence?". But before diving into this, we go back to the expression for the divergence:

$$D = \frac{2(\mu_B - \mu_G)^2}{\sigma_B^2 + \sigma_G^2} \tag{17}$$

It can be seen that the divergence is independent of the PD. Furthermore, if $\sigma_G = \sigma_B$, (17) simplifies to:

$$D = \frac{(\mu_B - \mu_G)^2}{\sigma^2}$$

From these remarks, the main question can be split into the following subquestions of increasing complexity:

1. Is the powerstat independent of the PD?
2. Is the mapping from powerstat to divergence one-to-one if the distributions have identical standard deviations?
3. Is the mapping from powerstat to divergence always one-to-one?

The main question can only be answered with "yes" if this is also the answer to all the subquestions. As the third subquestion is the same as the main question, the main question is automatically answered after answering all subquestions. This is done in the following subsections.

5.2.1 PD independence

In this subsection, we check whether the powerstat is independent of the PD. If this is not the case, we know for sure that a one-to-one mapping from powerstat to divergence does not exist. Recall that the powerstat is computed by:

$$PS = \frac{2A_{PC} - 1}{1 - PD} \tag{18}$$

Here, A_{PC} denotes the area under the power curve. The expression for A_{PC} is given in (16). Figure 24 shows a graphical representation of this area.

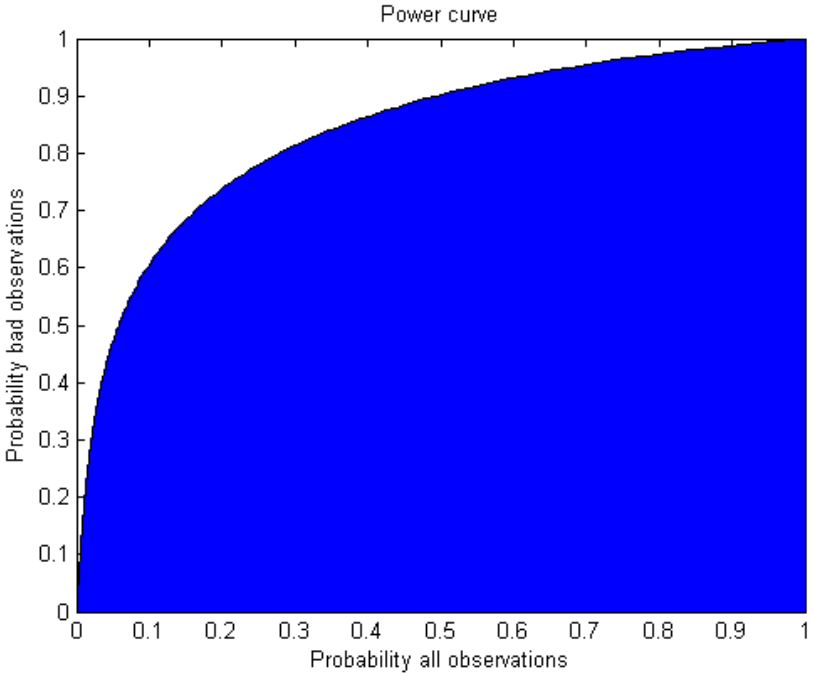

Figure 24: The blue area is the area under the power curve.

However, since we want to show that the powerstat is independent of the PD (which appears only in the distribution function of the model scores of all observations), it is better to write the probability of an arbitrary observation having a model score lower than (or equal to) s as a function of the probability of a bad observation having a model score lower than (or equal to) s. That is:

$$p_T = pc^{-1}(p_B) = F_T(F_B^{-1}(p_B)) \tag{19}$$

If we compute the area under $pc^{-1}(p_B)$ by integrating $pc^{-1}(p_B)$ with respect to p_B, we obtain the area as shown in Figure 25.

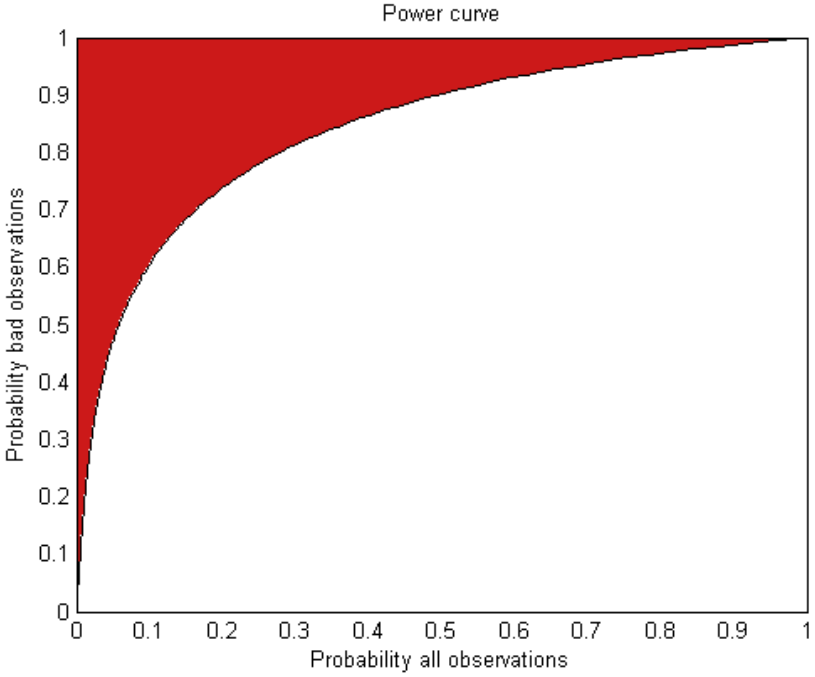

Figure 25: The red area is the area bounded by the power curve, the vertical axis, and the horizontal line $p_B = 1$.

Comparing Figure 24 and Figure 25, we can see that the blue area of Figure 24 can also be computed by subtracting the red area of Figure 25 from the total area. So, using (16) and (19) the area under the power curve can also be obtained by:

$$A_{PC} = 1 - \int_0^1 pc^{-1}(p_B)dp_B \tag{20}$$

We expect the PD-term in (18) to cancel out the effect of the PD in the integral computing the area under the power curve, as given in (20). We start with this area.

$$\begin{aligned} A_{PC} = 1 - \int_0^1 pc^{-1}(p_B)dp_B &= 1 - \int_0^1 F_T\left(F_B^{-1}(p_B)\right)dp_B \\ &= 1 - (1 - PD)\int_0^1 N(N^{-1}(p_B;\mu_B,\sigma_B);\mu_G,\sigma_G)dp_B \\ &\quad - PD\int_0^1 N(N^{-1}(p_B;\mu_B,\sigma_B);\mu_B,\sigma_B)dp_B \\ &= 1 - (1 - PD)\int_0^1 N(N^{-1}(p_B;\mu_B,\sigma_B);\mu_G,\sigma_G)dp_B - PD\int_0^1 p_B dp_B \\ &= 1 - (1 - PD)\int_0^1 N(N^{-1}(p_B;\mu_B,\sigma_B);\mu_G,\sigma_G)dp_B - \frac{1}{2}PD \end{aligned}$$

Plugging this in (15) gives:

$$PS = \frac{2A_{PC} - 1}{1 - PD} = \frac{1}{1 - PD}\left(1 - PD - 2\,(1 - PD)\int_0^1 N(N^{-1}(p_B; \mu_B, \sigma_B); \mu_G, \sigma_G)dp_B\right)$$
$$= 1 - 2 * \int_0^1 N(N^{-1}(p_B; \mu_B, \sigma_B); \mu_G, \sigma_G)dp_B \quad (21)$$

This shows that the powerstat is indeed independent of the PD. The first subquestion can therefore be answered with "yes".

5.2.2 Identical standard deviations

Now, we suppose the good and bad observations have the same standard deviation, σ. The mean can be different, however. The expression for the divergence can then be simplified to:

$$D = \frac{(\mu_B - \mu_G)^2}{\sigma^2}$$

For rewriting the expression for the powerstat in (21), the following basic rules for shifting to the standard normal distribution can be used.

$$N(x;\ \mu, \sigma) = N\left(\frac{x - \mu}{\sigma}; 0, 1\right)$$
$$N^{-1}(x;\ \mu, \sigma) = \sigma * N^{-1}(x; 0, 1) + \mu$$

The expression for the powerstat becomes:

$$PS = 1 - 2 * \int_0^1 N(N^{-1}(p_B; \mu_B, \sigma); \mu_G, \sigma)dp_B = 1 - 2 * \int_0^1 N\left(N^{-1}(p_B; 0{,}1) + \frac{\mu_B - \mu_G}{\sigma}; 0{,}1\right)dp_B$$

Since the normal distribution function is always positive and monotone increasing, it follows that the powerstat and the term $\varphi := (\mu_B - \mu_G)/\,\sigma$ can be mapped one-to-one. Furthermore, because $D = \varphi^2$, we know that $\varphi \Rightarrow D$ (i.e. φ implies D) and $D \Rightarrow |\varphi|$. Also, the normal density function is symmetric, so it follows that $PS(\varphi) = -PS(-\varphi)$. Therefore, the divergence implies the absolute value of powerstat. So if $\sigma_G = \sigma_B$, then $PS \Rightarrow D$ and $D \Rightarrow |PS|$.

In practical applications, the sign of the powerstat is known. In general, credit rating models are built in such a way that a higher model score corresponds to higher creditworthiness. Therefore, the powerstat is supposed to have a positive sign. We can therefore assume that the sign of the powerstat is known and therefore $D \Rightarrow |PS|$ can be generalized to $D \Rightarrow PS$. It follows that, if the standard deviations of the model score distributions are identical, there is a one-to-one relation between the powerstat and the divergence. So also the second subquestion can be answered with "yes".

5.2.3 General case

Now we suppose that $\sigma_G \neq \sigma_B$. The expression for the powerstat is then:

$$PS = 1 - 2 * \int_0^1 N\left(\frac{\sigma_B}{\sigma_G}N^{-1}(p_B; 0{,}1) + \frac{\mu_B - \mu_G}{\sigma_G}; 0{,}1\right)dp_B$$

For simplicity, define a and b such that: $a = \sigma_B/\sigma_G$ and $b = (\mu_B - \mu_G)/\sigma_G$. Then:

$$PS = 1 - 2 * \int_0^1 N(a * N^{-1}(p_B; 0{,}1) + b; 0{,}1)dp_B$$

The divergence can also be expressed in terms of a and b.

$$D = \frac{2(\mu_B - \mu_G)^2}{\sigma_B^2 + \sigma_G^2} = 2 * \frac{b^2}{a^2 + 1}$$

In order to prove that $PS \Rightarrow D$ and $D \Rightarrow |PS|$ also hold in the general case, we need to show that if $D_1 < D_2$ then $|PS_1| < |PS_2|$; and that if $|PS_1| < |PS_2|$, then $D_1 < D_2$.

First, the expression for the powerstat is transformed by applying the transformation $y = N^{-1}(p_B; 0,1)$. This yields:

$$\begin{aligned} PS &= 1 - 2 * \int_{-\infty}^{\infty} N(ay + b; 0,1) * \frac{1}{\sqrt{2\pi}}\, e^{-\frac{y^2}{2}} dy \\ &= 1 - 2 * N\left(b; 0, \sqrt{1 + a^2}\right) \\ &= \ 1 - 2 * N\left(\frac{b}{\sqrt{1 + a^2}}; 0,1\right) \end{aligned} \tag{22}$$

The second equality in the equation above follows from regarding the integral as an application of Bayes' theorem (Ross, 2010). That is, consider the independent variables Y and Z, both standard normally distributed with distribution function F and density function f. Then:

$$P(Z - aY \leq b) = \int_{-\infty}^{\infty} F_Z(ay + b) * f_Y(y) dy = \int_{-\infty}^{\infty} N(ay + b; 0,1) * \frac{1}{\sqrt{2\pi}}\, e^{-\frac{y^2}{2}} dy$$

It is known that a linear combination of independent normally distributed random variables is also normally distributed (Ross, 2010). Therefore, it follows that $Z - aY$ is normally distributed with mean 0 and standard deviation $\sqrt{1 + a^2}$. Therefore, $P(Z - aY \leq b)$ can also be written as $N\left(b; 0, \sqrt{1 + a^2}\right)$. This leads to the above expressions for the powerstat.

By the symmetry of the normal density function, the powerstat is negative for positive values of $b/\sqrt{1 + a^2}$ and positive for negative values of $b/\sqrt{1 + a^2}$. Also, from this symmetry it follows that $PS\left(b/\sqrt{1 + a^2}\right) = -PS\left(-b/\sqrt{1 + a^2}\right)$. It is therefore sufficient to only consider the positive powerstat.

Going back to what we need to show, we assume that there are values of a_1, a_2, b_1, and b_2, such that $D_1 < D_2$. This means that $-\sqrt{D_1} > -\sqrt{D_2}$, which in turn implies:

$$-\frac{|b_1|}{\sqrt{1 + a_1^2}} > -\frac{|b_2|}{\sqrt{1 + a_2^2}} \tag{23}$$

If we would plug this into (22), we see that $|PS_1| < |PS_2|$.

Conversely, if we assume $|PS_1| < |PS_2|$, then again (23) follows. Note that both the left and right side of the inequality are negative. Note that:

$$D = 2\left(-\frac{|b|}{\sqrt{1 + a^2}}\right)^2$$

From this, it follows that $D_1 < D_2$.

We showed that $D_1 < D_2 \Rightarrow |PS_1| < |PS_2|$ and that $|PS_1| < |PS_2| \Rightarrow D_1 < D_2$, so $|PS| \Leftrightarrow D$. This is the desired result. It has now been proved that – under the assumptions made – $PS \Rightarrow D$ and $D \Rightarrow |PS|$.

Table 7 illustrates the mapping between the powerstat and the divergence.

Powerstat		Divergence
Positive	Negative	
0	0	0
0.1	-0.1	0.032
0.2	-0.2	0.128
0.3	-0.3	0.297
0.4	-0.4	0.550
0.5	-0.5	0.910
0.6	-0.6	1.417
0.7	-0.7	2.148
0.8	-0.8	3.285
0.9	-0.9	5.411
1	-1	∞

Table 7: Values for the powerstat and the corresponding values of the divergence.

5.3 Applications of the equivalence

As shown in the previous section, the performance measures powerstat and divergence are equivalent. The divergence is calculated from a nice tractable formula involving the means and variances, so one can expect the it is possible to calculate its confidence interval. Because of the equivalence between powerstat and divergence, we can now create a confidence interval for the powerstat as well.

Also, the equivalence of the powerstat and divergence makes way for another welcome addition to the current use of the powerstat. The powerstat is often used in practice, but it is difficult to interpret its meaning. In the previous section we found an alternative expression for the powerstat, which makes the interpretation easier.

We will elaborate on these two points in the following subsections.

5.3.1 Creating the confidence interval

In the above we assumed that the number of observations was (close to) infinite. However, in practice we are always dealing with a limited sample of observations. In that case, the computed powerstat and divergence are – to some extent – dependent on the sample. It therefore makes sense to compute a confidence interval around the powerstat or divergence value. That way, the uncertainty in the value for the powerstat or the divergence is quantified. Since the expression for the divergence is very clear-cut, we first find a confidence interval for this measure. Using the one-to-one mapping from the divergence to the powerstat, we can then compute a confidence interval for the powerstat as well.

Recall that the expression for divergence is:

$$D = \frac{2(\mu_B - \mu_G)^2}{\sigma_B^2 + \sigma_G^2}$$

Creating a confidence interval for the divergence is not very straightforward. Instead, we first find the confidence interval for $\pm\sqrt{D/2}$, so for

$$\frac{\mu_B - \mu_G}{\sqrt{\sigma_B^2 + \sigma_G^2}} =: SN$$

(The given name SN stems from the similarity with the signal-to-noise ratio, SNR (Lupton, Gunn, & Szalay, 1999).) The variances of the model scores for the good observations and the model scores of the bad observations, σ_G^2 and σ_B^2 respectively, can be estimated by s_G^2 and s_B^2, respectively, which are given by:

$$s_G^2 = \frac{1}{N_G - 1}\sum_{i=1}^{N_G}(x_{i,G} - \overline{x}_G)^2$$

$$s_B^2 = \frac{1}{N_B - 1}\sum_{j=1}^{N_B}(x_{j,B} - \overline{x}_B)^2$$

Here, N_B and N_G are the number of bad and good observations, respectively. Furthermore, $\overline{x}_G$ and $\overline{x}_B$ denote the sample means, where $x_{i,G}, i = 1 \dots N_G$ and $x_{j,B}, j = 1 \dots N_B$ are the model scores for the good and bad observations, respectively.

As s_G^2 and s_B^2 can be found directly from the data, it is more convenient to find the confidence interval for $\widehat{SN}$ first, where $\widehat{SN}$ is defined as:

$$\widehat{SN} := \frac{\mu_B - \mu_G}{\sqrt{s_B^2 + s_G^2}}$$

Since s_G^2 and s_B^2 are consistent estimators, they almost surely converge to σ_G^2 and σ_B^2 (Taboga, 2012). The confidence interval for $\widehat{SN}$ will therefore give a good estimate for the confidence interval for SN.

The expression for $\widehat{SN}$ is closely related to the test statistic in Welch's t-test (Welch, 1947). For this test statistic a distribution is known, which we can use in constructing the confidence interval for $\widehat{SN}$. With Welch's t-test, one can test whether the means of observations from two independent groups are the same. The standard deviations are not assumed to be equal. Because of this, the number of degrees of freedom is computed in a slightly complicated manner.

The test statistic for Welch's t-test is given by:

$$\frac{(\overline{x}_B - \overline{x}_G) - (\mu_B - \mu_G)}{\sqrt{s_B^2/N_B + s_G^2/N_G}}$$

The sample difference between the mean of the good model scores and the mean of the bad model scores is given by $\overline{x}_B - \overline{x}_G$. Welch's statistic follows a t-distribution with the number of degrees of freedom (df) computed as follows (Pan, 2002):

$$df = \frac{\left(\frac{s_B^2}{N_B} + \frac{s_G^2}{N_G}\right)^2}{\frac{s_B^4}{N_B^2(N_B - 1)} + \frac{s_G^4}{N_G^2(N_G - 1)}}$$

Using the t-distribution, we can then fix a confidence level α (for example $\alpha = 95\%$) such that:

$$P\left(\frac{\overline{x}_B - \overline{x}_G}{\sqrt{\frac{s_B^2}{N_B} + \frac{s_G^2}{N_G}}} - t_{(1-\alpha)/2,df} \leq \frac{\mu_B - \mu_G}{\sqrt{\frac{s_B^2}{N_B} + \frac{s_G^2}{N_G}}} \leq \frac{\overline{x}_B - \overline{x}_G}{\sqrt{\frac{s_B^2}{N_B} + \frac{s_G^2}{N_G}}} + t_{(1-\alpha)/2,df}\right) = \alpha$$

Here, $t_{(1-\alpha)/2,df}$ is the critical value for a t-distribution with df degrees of freedom. This can be rewritten such that:

$$P\left(\frac{\overline{x}_B - \overline{x}_G}{\sqrt{s_B^2 + s_G^2}} - t_{(1-\alpha)/2,df}\frac{\sqrt{\frac{s_B^2}{N_B} + \frac{s_G^2}{N_G}}}{\sqrt{s_B^2 + s_G^2}} \leq \widehat{SN} \leq \frac{\overline{x}_B - \overline{x}_G}{\sqrt{s_B^2 + s_G^2}} + t_{(1-\alpha)/2,df}\frac{\sqrt{\frac{s_B^2}{N_B} + \frac{s_G^2}{N_G}}}{\sqrt{s_B^2 + s_G^2}}\right) = \alpha$$

This yields a α-confidence interval for $\widehat{SN}$. It can be seen that an increase in the number of observations decreases the width of the interval. An infinite number of observations would yield an "interval" of width 0, thus consisting of only $\widehat{SN}$. Because of the presence of N_B and N_G in the expressions for the boundaries, it follows that the width of the confidence interval also depends on the observed default frequency.

Since $\widehat{SN}$ converges to SN, the confidence interval for SN can be approximated by the one for $\widehat{SN}$. In turn, the confidence interval for the divergence can be obtained from that of the SN, since we know that $D = 2 * SN^2$. Set $a_{low,SN}$ as the lower bound and $a_{upp,SN}$ as the upper bound obtained for SN. As the mapping from SN to the divergence is not one-to-one (due to the squared value of SN), some care needs to be taken in transforming the confidence interval for SN to a confidence interval for the divergence. Three cases can be identified.

1. Both $a_{low,SN} < 0$ and $a_{upp,SN} \leq 0$;
2. Both $a_{low,SN} \geq 0$ and $a_{upp,SN} > 0$;
3. $a_{low,SN} < 0$ and $a_{upp,SN} > 0$.

For the first two cases, the lower bound is given by $a_{low,D} = \min(2a_{low,SN}^2, 2a_{upp,SN}^2)$ and the upper bound by $a_{upp,D} = \max\left(2a_{low,SN}^2, 2a_{upp,SN}^2\right)$. This upper bound is the same for the third case, but the lower bound is then $a_{low,D} = 0$, as 0 lies within this third confidence interval.

From (22), the powerstat can be expressed in terms of SN as well:

$$PS = 1 - 2N(SN; 0,1) \tag{24}$$

It follows from this expression that SN can be mapped one-to-one to the powerstat. Because of this, the confidence interval for the powerstat follows easily from the confidence interval for SN. By plugging in the lower and upper bound for the SN-interval, the upper and lower bound for the powerstat interval, respectively, can be found. In Figure 26 the width of the confidence interval for the powerstat is plotted against the number of observations and the PD.

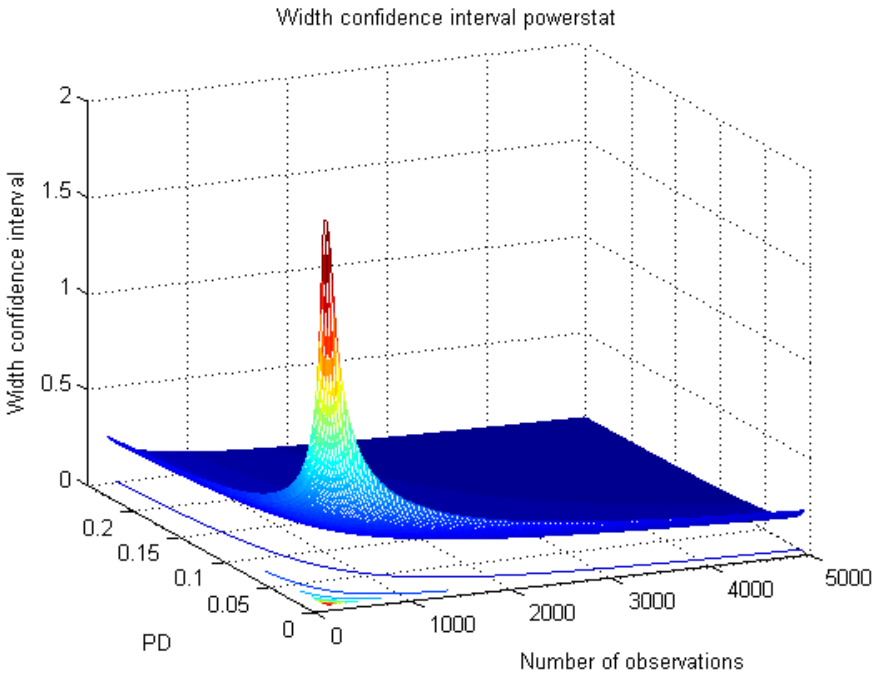

Figure 26: The width of the confidence interval for the powerstat, plotted against the number of observations and the PD (which is assumed to be the same as the observed default frequency).

For clarity, the corresponding contour plot is given separately in Figure 27.

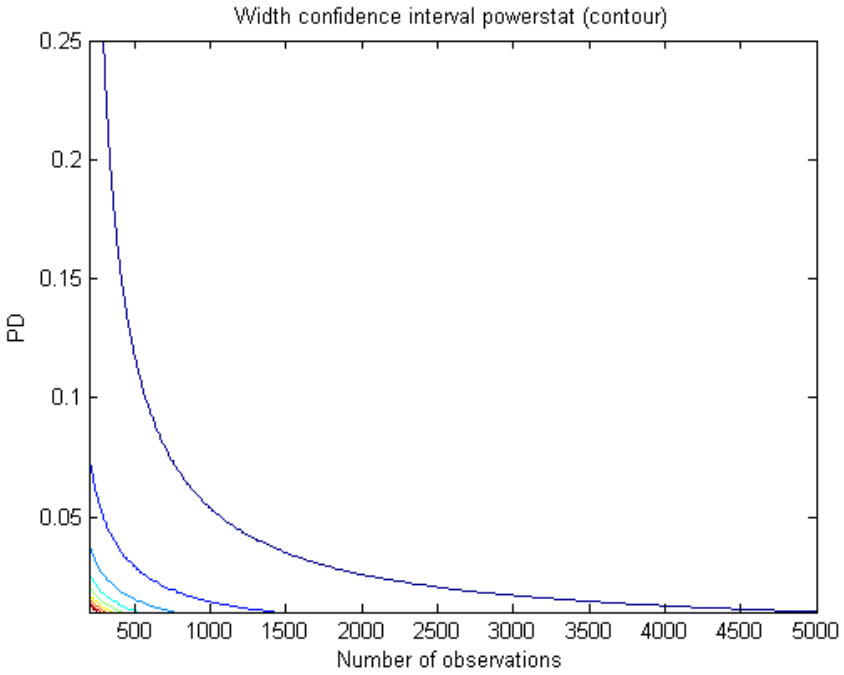

Figure 27: Contour plot for the width of the confidence interval for the powerstat, plotted against the number of observations and the PD (which is assumed to be the same as the observed default frequency).

It can be seen from Figure 26 and Figure 27 that the width of the interval decreases if the number of observations increases or if the PD increases. If the PD increases, there are

relatively more bad observations that can be used for model estimation. The powerstat can therefore be estimated better, leading to a smaller width of its confidence interval.

5.3.2 Interpretation of the powerstat

Recall that the powerstat is a measure used to check the discriminatory power of a model. For computing the powerstat, the area under the power curve is needed. The power curve is obtained by plotting the distribution function of the model scores of the bad observations against the distribution function of the model scores of all observations. The powerstat is then computed from (18).

If we can assume that the model scores of the good and bad observations are both normally distributed, we showed in (24) that the expression for the powerstat can be rewritten to:

$$PS = 1 - 2N\left(\frac{\mu_B - \mu_G}{\sqrt{\sigma_B^2 + \sigma_G^2}}; 0, 1\right)$$

Using the identity $N((x-\mu)/\sigma; 0,1) = N(x; \mu, \sigma)$, this can in turn be rewritten to:

$$PS = 1 - 2N\left(0;\ \mu_G - \mu_B, \sqrt{\sigma_G^2 + \sigma_B^2}\right) \tag{25}$$

The second term containing the normal probability can be interpreted as the probability of the model score of a bad observation being higher than the model score of a good observation. This is not desirable, so this term should be as small as possible. The smaller this probability, the larger the powerstat, as can be seen from (25). Figure 28 shows the relevant distributions.

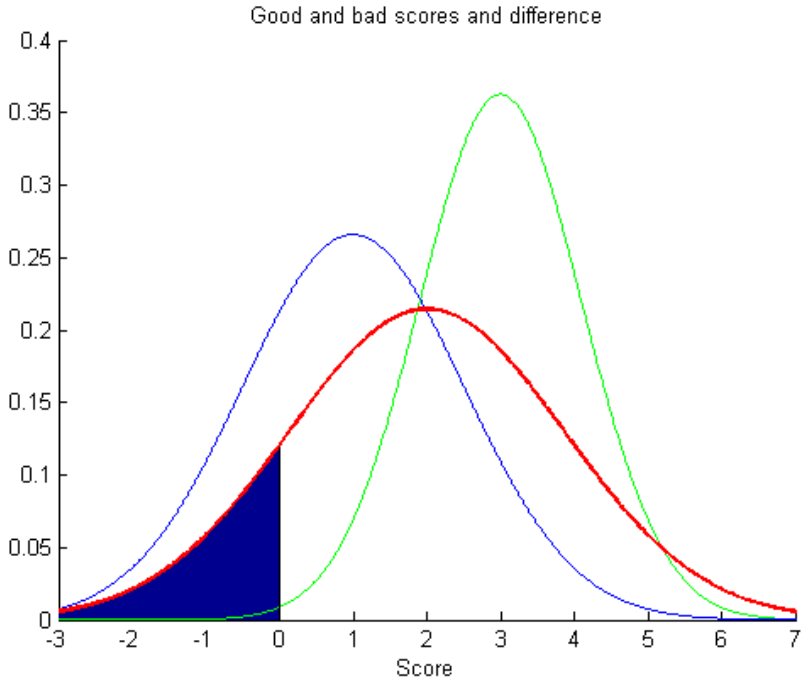

Figure 28: The blue graph is the distribution of the model scores of the bad observations, the green is that of the model scores of the good observations, and the red graph is the difference distribution. The filled area represents the probability of a bad observation having a higher model score than a good observation. For this figure we set: $\mu_G = 3$, $\mu_B = 1$, $\sigma_G = 1.1$, and $\sigma_B = 1.5$.

In Figure 28, the distributions of the model scores of the good and bad observations are shown in green and blue, respectively. Also, the difference function of the two distributions is plotted (red). If the difference in model scores is negative, the model score of a bad observation is higher than the model score of a good observation. The filled area therefore represents the probability of a bad observation having a higher model score than a good observation.

If the distributions of the model scores of the good and bad observations would be the same, then the probability of a bad observation having a higher model score than a good observation, would be exactly ½. That means, that a probability higher than ½ follows from a model that consequently rates bad observations higher than good observations. To "punish" for this behaviour, we may compute the measure $1/2 - P(s_B > s_G)$, which is negative in case the model rates bad observations higher than good observations. This measure ranges from – ½ to + ½ . Since it is more insightful to have a measure ranging from – 1 to +1, this measure can be calibrated to:

$$1 - 2P(s_B > s_G) = 1 - 2N\left(0;\ \mu_G - \mu_B, \sqrt{\sigma_G^2 + \sigma_B^2}\right) \tag{26}$$

This is exactly (25), the expression for the powerstat.

Using $P(s_B > s_G) = 1 - P(s_B < s_G)$ we can rewrite (26) as:

$$PS = 1 - 2P(s_B > s_G) = 2P(s_B < s_G) - 1 \tag{27}$$

It follows that the powerstat can also be interpreted as twice the probability of a bad observation having a lower model score than a good observation, minus 1. The powerstat can thus be seen as a linear transformation of the probability of a bad observation having a lower model score than a good observation. This direct relation between the powerstat and the probability of a bad observations having a lower model score than a good observation, is a new addition to the interpretation of the powerstat.

Table 8 shows several powerstat values and the corresponding probabilities of a bad observation having a lower model score than a good observation.

Powerstat	$P(s_B < s_G)$	Powerstat	$P(s_B < s_G)$
-1	0	0.1	0.55
-0.9	0.05	0.2	0.60
-0.8	0.10	0.3	0.65
-0.7	0.15	0.4	0.70
-0.6	0.20	0.5	0.75
-0.5	0.25	0.6	0.80
-0.4	0.30	0.7	0.85
-0.3	0.35	0.8	0.90
-0.2	0.40	0.9	0.95
-0.1	0.45	1	1
0	0.50		

Table 8: Powerstat values and the corresponding probability of a bad observation having a lower model score than a good observation.

Suppose for example that we have two models; for model 1 $P(s_{B,1} < s_{G,1}) = 0.75$ and for model 2 $P(s_{B,2} < s_{G,2}) = 0.80$. Since $P(s_B < s_G) = 1$ would hold for a model with perfect discriminatory power, we expect the second model to have more discriminatory power than the first. From (27) (and Table 8) it follows that $PS_1 = 0.50$ and $PS_2 = 0.60$, which indeed confirms that model 2 has more discriminatory power than model 1.

5.4 Shadow-Bond powerstat

An alternative for the Good-Bad modelling approach, is the Shadow-Bond approach. For the Good-Bad approach, the response variable of the observations is binary: an observation is either good (non-default) or bad (default). On the other hand, when the Shadow-Bond approach is used, the response variable is the rating category that an external rating agency classifies the observation in. The response variable is still discrete, but with more than two possibilities.

Throughout this chapter, only the Good-Bad approach is considered. There are two main reasons why the method for creating a confidence interval for the powerstat cannot be applied to the Shadow-Bond approach, these are explained in 5.4.2. But first, the computation of the powerstat for the Shadow-Bond approach is discussed in subsection 5.4.1.

5.4.1 Powerstat computation for Shadow-Bond approach

For the computation of the powerstat for both the Good-Bad approach and the Shadow-Bond approach three curves are needed: the power curve, the curve of the random model, and the curve for the crystal ball model. Remember that the powerstat is defined as the ratio of the area between the power curve and the curve of the random model and the area between the curve of the crystal ball model and the random model. See also Figure 29.

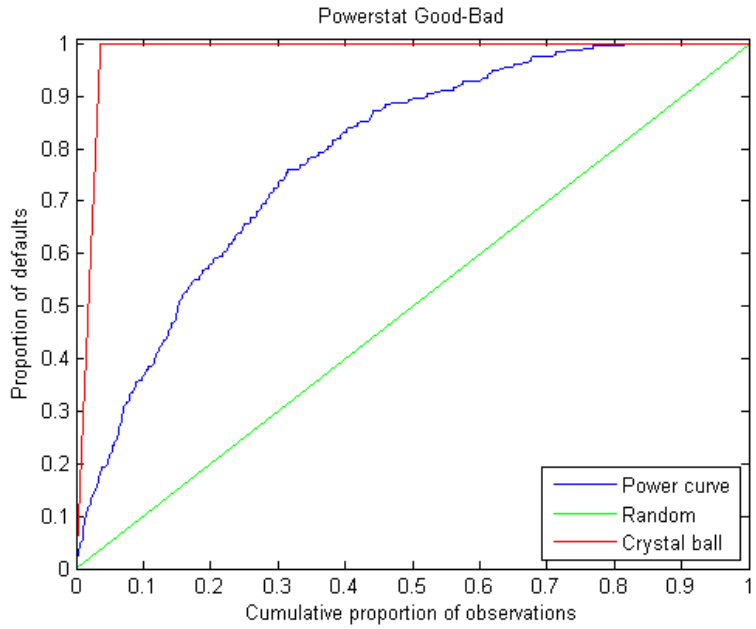

Figure 29: The power curve, the curve of the random model, and the curve of the crystal ball model, using the Good-Bad approach.

When using the Good-Bad approach, for each score s the proportion of bad observations with a lower model score than s is plotted against the proportion of all observations with a model score lower than s.

As there are more than two response categories when the Shadow-Bond approach is used, these curves should be created differently. When using the Shadow-Bond approach, for each score s the sum of the PDs (associated with the external ratings) of observations with a lower model score than s, divided by the sum of the PDs of all observations, is plotted against the proportion of observations with a model score lower than s. The rest of the computation of the powerstat is identical to that for the Good-Bad approach.

5.4.2 No confidence interval for Shadow-Bond approach

There are two main reasons that the proof of this chapter, leading to a confidence interval for the powerstat under the Good-Bad approach, cannot be extended to the Shadow-Bond approach. These reasons will be discussed in this subsection.

The creation of a confidence interval for the powerstat for the Good-Bad approach, required proving the one-to-one relation between the absolute value of the powerstat and the divergence. The expression for the divergence is:

$$D = 2\frac{(\mu_B - \mu_G)^2}{\sigma_B^2 + \sigma_G^2}$$

The subscripts B and G refer to the model scores of the bad observations and the model scores of the good observations, respectively. In this case, there are only two groups of observations: the "goods" and the "bads". For the Shadow-Bond approach, there are more groups. External rating agency Standard & Poor's would, for example, identify 22 different rating categories (Standard & Poor's, 2011). From the definition of the divergence, it is not clear how this increased number of groups would be incorporated. We may sum all pairwise divergences, or multiply all pairwise divergences, or sum only the pairwise divergences for the consecutive groups, etc. So since the Shadow-Bond divergence is not clearly defined, the divergence and powerstat cannot be linked for the Shadow-Bond approach, and the proof of section 5.2 cannot be extended to the Shadow-Bond powerstat.

Another reason for the difficulties in finding a confidence interval for the powerstat for the Shadow-Bond approach can be found by considering the difference between the powerstat for the Good-Bad approach and the powerstat for the Shadow-Bond approach. Figure 30 shows the graphs for creating the powerstat for the Shadow-Bond approach.

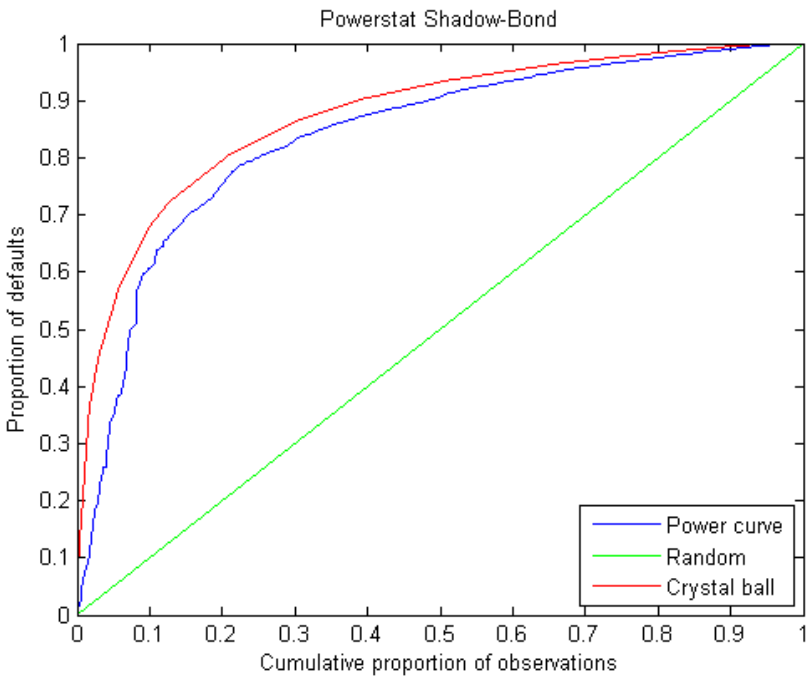

Figure 30: The power curve, the curve of the random model, and the curve of the crystal ball model, using the Shadow-Bond approach.

When we compare Figure 29 and Figure 30, it can be seen at first glance that the curves for the crystal ball model for both approaches are very different. The curve of the crystal ball model for the Good-Bad approach consists of two parts: a line with a very steep incline and a horizontal line. On the other hand, the curve of the crystal ball model for the Shadow-Bond approach is quite close to the power curve and is smoother than its Good-Bad counterpart. Because of this, the area between the power curve and the curve of the crystal ball model for the Shadow-Bond approach is much less than this same area for the Good-Bad approach. Recall that the powerstat is computed as the ratio of two areas: the area between the power curve and the curve of the random model, and the area between the curve of the crystal ball model and the curve of the random model. Thus, it follows that the powerstat is in general higher for the Shadow-Bond approach than for the Good-Bad approach. However, models based on the Shadow-Bond approach do not necessarily have a higher discriminatory power than models based on the Good-Bad approach. Therefore, the value of the powerstat should be interpreted differently according to the approach used.

This suggests that the powerstat for the Shadow-Bond approach and the powerstat for the Good-Bad approach can in fact be interpreted as two different measures for the discriminatory power of a model. Extending properties (such as the confidence interval) from one to the other may therefore not be as straightforward as one can expect based on the similar names.

6 Powerstat & divergence – Application

In the previous chapter we developed a method to compute a confidence interval for the powerstat: a measure for the discriminatory power of a model. The better a rating model can discriminate between good and bad observations, the higher its powerstat. But from a statistical point of view, a single powerstat value contains little information. It is therefore desirable to create a confidence interval for the powerstat, which increases the understanding of the uncertainty of this powerstat value. In this chapter, we will provide numerical background for the theory of the previous chapter.

First, we recapitulate how the confidence interval for the powerstat is created. In order to create a confidence interval for the powerstat (PS), we first needed to link it to another measure for the discriminatory power of a model: the divergence (D). This is done by linking both measures to a third one: SN. This SN is given by:

$$SN = \frac{\mu_B - \mu_G}{\sqrt{\sigma_B^2 + \sigma_G^2}}$$

We derived the following two relations:

$$D = 2 * SN^2$$
$$PS = 1 - 2 * N(SN; 0,1) \tag{28}$$

As s_B and s_G converge to σ_B and σ_G, respectively (Taboga, 2012), we approximated SN by $\widehat{SN}$:

$$\widehat{SN} = \frac{\mu_B - \mu_G}{\sqrt{s_B^2 + s_G^2}}$$

By using Welch's t-test (Welch, 1947), a confidence interval for $\widehat{SN}$ is created:

$$\frac{\overline{x}_B - \overline{x}_G}{\sqrt{s_B^2 + s_G^2}} - t_{(1-\alpha)/2,df} \frac{\sqrt{\frac{s_B^2}{N_B} + \frac{s_G^2}{N_G}}}{\sqrt{s_B^2 + s_G^2}} \leq \widehat{SN} \leq \frac{\overline{x}_B - \overline{x}_G}{\sqrt{s_B^2 + s_G^2}} + t_{(1-\alpha)/2,df} \frac{\sqrt{\frac{s_B^2}{N_B} + \frac{s_G^2}{N_G}}}{\sqrt{s_B^2 + s_G^2}}$$

As $\widehat{SN}$ also converges to SN (which follows from the definition of $\widehat{SN}$ and the convergence of s_B and s_G), this confidence interval for $\widehat{SN}$ can be used as confidence interval for SN. By using (28), the confidence interval for the powerstat can also be obtained.

In order to use the divergence, we had to assume that the model scores for the good and bad observations are normally distributed. In 6.1 we check whether this is indeed the case. Also, in 5.3.1 we created the confidence interval for the powerstat. In 6.2 we use a numerical approach to confirm that this is indeed a confidence interval for the powerstat. To give an example of this confidence interval, we use the development data of a recently developed model (the Poland SME PD model) in 6.3. For Rabobank, the Matlab implementation of this method is explained in Appendix F.

6.1 Normality of the model scores

In order to find an equivalence relation between the powerstat and the divergence, we had to base the powerstat on the same assumptions as the divergence. Therefore, both the model scores of the bad observations and the model scores of the good observations are assumed to be normally distributed. In order to apply the results of this equivalence relation to the

models used in practice, it is useful to check if the model scores of the good observations and the model scores of the bad observations in a real development dataset are indeed (close to) normally distributed.

Note that the method of the previous chapter only applies to the computation of the discriminatory power of *models*, not individual factors. These factors are often not normally distributed, so the derivation of the previous chapter would not apply.

For testing the normality of the model scores, three different development datasets are available. All three were used in developing a rating model for Rabobank using the Good-Bad approach. For each dataset, we have two sets of model scores: the model scores for the good observations and the model scores for the bad observations. In order to test whether these model scores are normally distributed, a Jarque-Bera test is performed (Jarque & Bera, 1980). This test uses the skewness (γ) and the kurtosis (κ) of the empirical distribution and compares this with the skewness and kurtosis of a normal distribution. The Jarque-Bera test statistic (JB) is:

$$JB = \frac{N}{6}\left(\gamma^2 + \frac{(\kappa - 3)^2}{4}\right)$$

If N (the number of observations) is large enough, JB follows a chi-squared distribution with two degrees of freedom. If we set the confidence level at 95%, this means that the critical value for JB is 5.99. For the three development datasets, the value for JB is given in Table 9.

Development dataset	Good / Bad	JB value
1	Good	88.81
	Bad	0.874
2	Good	544.76
	Bad	29.49
3	Good	24.82
	Bad	1.746

Table 9: The values for the Jarque-Bera test statistic for three different datasets.

From Table 9 we can see that $JB > 5.99$ in four cases. This means that in those cases we can reject – with 95% certainty – the null hypothesis that the model scores are normally distributed.

This would suggest that the results from the previous chapter cannot be used in practice. To see if there is some lenience in this, we plot the each empirical model score distribution together with a normal distribution. This normal distribution is the normal distribution that has the same mean and standard deviation as the empirical model score distribution. These plots can be found in Appendix G.

It can be seen that the model score distributions deviate most from the normal distributions around the extreme model scores. An important reason for this, is that the model scores are bounded, as they are a linear combination of factor scores that all have a bounded range of $[0,10]$. Other than that, the empirical distributions follow the normal distribution quite closely. Therefore, the results of the previous chapter can still be applied to practical data, provided that they are used with caution.

For extreme powerstat values (close to -1 or 1) in particular, the confidence interval may not hold. That is because these extreme powerstat values are almost fully determined by the tail behaviour of the distributions for the model scores of the good and bad observations. And as we noticed, the empirical model score distributions deviate most from the normal distributions around the extreme model scores.

6.2 Checking the confidence interval

In 5.3.1 we developed a confidence interval for the powerstat. To check whether the constructed interval is indeed the interval in which the powerstat lies with a confidence level of α (predetermined, here we will use $\alpha = 95\%$), we use a Monte Carlo method. For this method, a large number of datasets is generated. Each dataset consists of good and bad observations with model scores that are generated from normal distributions. For each dataset the 95% "confidence" interval for the powerstat can be computed, using the method described in 5.3.1. As it needs to be checked whether this interval is really a confidence interval, we will refer to it as "interval".

In order to check whether the interval is a confidence interval for the powerstat, the "real" powerstat needs to be computed. This can be done, as the underlying model parameters are known. Also, the interval described in 5.3.1 is computed. Note that for this computation only $\bar{x}_B$, $\bar{x}_B$, s_B, s_G, N_B, and N_G are required, which are all available for each individual dataset. Then, we check for each dataset whether the powerstat lies within the interval. If the interval is indeed a 95% confidence interval, the expectation is that in 95% of the datasets the powerstat will lie in the interval.

This Monte Carlo check is performed for different values of the powerstat. These different powerstat values are created by varying $\mu_B - \mu_G$. Each Monte Carlo check is performed on 100000 datasets of 5000 observations. The results for negative powerstat values are similar to those for the positive powerstat values. Therefore, only the results for the positive powerstat values are shown.

Powerstat	% of dataset intervals containing the powerstat	P-value of outcome when: H_0: $P(PS\ in\ interval) = 95\%$
0	94.99%	89%
0.1	94.96%	57%
0.2	95.08%	24%
0.3	94.95%	47%
0.4	94.97%	67%
0.5	95.01%	88%
0.6	95.02%	77%
0.7	95.05%	46%
0.8	94.99%	89%
0.9	94.93%	31%

Table 10: For different powerstat values, we determine for how many datasets the interval contained the powerstat. Also, the p-value for the observed outcomes is given.

It can be seen from Table 10 that the values in the second column are very close to 95%. Whether this is close enough to conclude that the confidence level is indeed 95%, can be checked statistically. Each Monte Carlo dataset can be considered as an independent binomial test. There are two possible outcomes: either the powerstat lies within the interval constructed on the dataset (success), or outside of it (failure). We can define the null hypothesis as:

$$H_0: P(PS\ in\ interval) = 95\%$$

The alternative hypothesis is subsequently defined as:

$$H_1: P(PS\ in\ interval) \neq 95\%$$

Then, the p-values can be computed for the observed number of successes out of 100000 tests. These are given in the third column of Table 10. As the p-values are all large enough, it can be concluded that the interval developed in 5.3.1 indeed yields a confidence interval for the powerstat.

6.3 Confidence interval example

To get some more feel for the powerstat and its confidence interval, we compute these values for a recently developed model; the Poland SME PD model. This model develops a rating system for the probability of default for small and medium enterprises in Poland. The observed default frequency is between 1% and 5%. The powerstat for this model is computed to be 0.5901, which is quite high for a model developed using the Good-Bad approach. We vary the number of observations – while keeping the distribution parameters fixed – to create different confidence intervals. The confidence level is fixed at 95%. The results are given in Table 11.

Number of observations	Lower bound	Upper bound	Width interval
150	0.2261	0.8265	0.6004
500	0.4039	0.7365	0.3326
1000	0.4622	0.6979	0.2357
2000	0.5016	0.6685	0.1669
5000	0.5353	0.6409	0.1056
10000	0.5517	0.6264	0.0747

Table 11: Lower and upper boundaries for the 95%-confidence interval for the powerstat (0.5901) for a varying number of observations. The width of the interval is also given.

Increasing the number of observations also increases the lower bound and, at the same time, the upper bound decreases. This narrows the width of the confidence interval. It can be seen that for 500 observations, the confidence interval was 0,3326 wide. For 2000 observations, the width decreased to 0,1669. Note however, that while the number of observations increased by a factor 4, the interval width was only cut in half. This effect is due to the $1/\sqrt{N}$ factor in the expression for the interval bounds for SN. Furthermore, the non-linear transformation $SN \rightarrow PS$ is the reason that $0{,}1669/0{,}3326 \neq 1/2$.

7 Conclusions

In this chapter we give some concluding remarks on the methods developed in this thesis: the ellipse method for quantifying the uncertainty in the factor coefficients found by regression, and the method for deriving a confidence interval to quantify the uncertainty in the powerstat. Also, we give some suggestions for further research.

7.1 Conclusions for the ellipse method

A credit rating model uses a number of characteristics of the counterparty as input. These are called the *factors*. Some factors are more important than others in estimating the creditworthiness of a counterparty, and therefore get a higher coefficient in the model. These coefficients are estimated on a dataset. The ellipse-method describes a method to create a two-dimensional (ellipse-shaped) confidence region for each factor coefficient, where the second additional dimension represents the combination of all other coefficients.

With this ellipse-shaped region two intervals can be found per factor coefficient, both centred around the coefficient that is estimated by regression. The inner interval shows how much the coefficient is allowed to change, if all other coefficients should remain unchanged. On the other hand, the outer interval shows the possible changes for the factor coefficient, if at least one other coefficient is allowed to change as well.

Using numerical tests, we were able to ascertain that the ellipse is indeed a confidence region. This was also the case for the ellipse for the coefficients derived by logistic regression, where the covariance matrix had to be approximated.

From the example data we saw that in practice the difference between the inner and outer interval was relatively small. This is a consequence of the relatively high number of factors included in the model and the low correlation between the factors. Then, the correlation between one coefficient and the sum of all other coefficients is quite small. We showed that the ratio of the widths of the outer and inner intervals is directly related to this correlation, where a small correlation yields a ratio close to 1 (similar widths).

7.2 Conclusions for powerstat & divergence

When testing the performance of a credit rating model, very often the powerstat is used for measuring its discriminatory power. But as this powerstat can only be computed based on a finite number of observations, it makes sense to construct a confidence interval for this value to increase the understanding of it.

However, a confidence interval for the powerstat cannot be created directly from its definition. Instead, we use another measure for the discriminatory power: the divergence. This divergence is computed from a tractable formula based on only the means and the variances, and therefore we are able to construct its confidence interval. By linking the powerstat to the divergence, we can extend the properties of the divergence, including the confidence interval, to the powerstat.

We first showed that – just like the divergence (D) – the powerstat (PS) is independent of the PD. Next, starting with a special case, we showed that the following relations hold in general: $PS \Rightarrow D$ and $D \Rightarrow |PS|$. Since the sign of the powerstat is in practice always positive, these relations imply a one-to-one mapping from the powerstat to the divergence.

An additional application of the equivalence between the powerstat and the divergence resulted in a new interpretation of the powerstat. It turned out that the powerstat is linearly related to the probability of a good observation having a higher model score than a bad observation.

In order to link the powerstat to the divergence, we have to assume that the model scores of the good observations and those of the bad observations are both normally distributed. Using a real development dataset, we checked this assumption for the model scores of the observations. It followed that in general, the empirical distributions of the model scores do not deviate too much from a normal distribution. The exceptions are in the tails of the distributions: for extreme model scores, the empirical distributions are not comparable to normal distributions. A reason for this is that the model scores are bounded. Since the confidence intervals of extremely high or low powerstat values (close to -1 or 1) are strongly influenced by the tail behaviour of the model score distributions, we suggest that the confidence interval is used with caution for these powerstat values.

7.3 Suggestions for further research

In order to extend the knowledge on the topics considered in this thesis, we have some suggestions for further research.

The ellipse for coefficients derived by logistic regression was based on an approximation of the covariance matrix. Even though we showed in 4.1 that the computed ellipses are indeed confidence regions and therefore perform well, further research may yield an exact covariance matrix. This may improve the confidence region as well.

Furthermore, during the discussion of the ellipse method the primary focus was on providing more insight in the situation of changing *one* factor coefficient. To extend this method to changing multiple coefficients simultaneously, more research is needed.

As was pointed out in 5.4, a confidence interval could only be created for the powerstat of a model based on the Good-Bad approach, not for the powerstat of a model based on the Shadow-Bond approach. Since quite a number of models is based on the Shadow-Bond approach, further research on the creation of a confidence interval for the Shadow-Bond powerstat may be useful.

8 Bibliography

Agresti, A. (2002). *Categorical Data Analysis.* John Wiley & Sons, Inc.

Altman, E. I. (1968, September). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The journal of finance, XXIII*(4), 589-609.

Altman, E. I. (2006). *Default Recovery Rates and LGD in Credit Risk Modeling and Practice: An Updated Review of the Literature and Empirical Evidence.* Working paper.

Basel Committee on Banking Supervision. (2000). *Principles for the Management of Credit Risk.* Bank for International Settlements.

Basel Committee on Banking Supervision. (2006). *International Convergence of Capital Measurement and Capital Standards, A Revised Framework Comprehensive Version.* Bank for International Settlements.

Bessis, J. (2011). *Risk Management in Banking* (3rd ed.). John Wiley & Sons Ltd.

Engelmann, B., Hayden, E., & Tasche, D. (2003). *Testing rating accuracy.* RISK.

Gini, C. (1912). *Variabilità e mutabilità.* Bologna.

Heij, C., De Boer, P., Franses, P. H., Kloek, T., & Van Dijk, H. K. (2004). *Econometric Methods with Applications in Business and Economics.* Oxford University Press.

Hosmer, D. W., & Lemeshow, S. (2000). *Applied Logistic Regression* (2nd ed.). John Wiley & Sons, Inc.

Hull, J. C. (2010). *Risk Management and Financial Institutions* (2nd ed.). Pearson.

Jarque, C. M., & Bera, A. K. (1980). Efficient tests for normality, homoscedasticity and serial independence of regression residuals. *Economics Letters, 6*(3), 255-259.

Jolliffe, I. T. (2002). *Principal Component Analysis* (2nd ed.). Springer.

Klecka, W. R. (1980). *Discriminant Analysis.* SAGE Publications, Inc.

Lupton, R. H., Gunn, J. E., & Szalay, A. S. (1999, September). A modified magnitude system that produces well-behaved magnitudes, colors, and errors even for low signal-to-noise ratio measurements. *The Astronomical Journal, 118*, 1406-1410.

Merton, R. (1974). On the Pricing of Corporate Debt: The Risk Structure of Interest Rates. *Journal of Finance, 29*, 449-470.

Pan, W. (2002). A comparative review of statistical methods for discovering differentially expressed genes in replicated microarray experiments. *Bioinformatics*, 546-554.

Rabobank. (2006). *General checklist PD models*.

Rabobank. (2008). *Mapping Rabobank Risk Ratings to S&P-ratings*.

Rabobank. (2010). *QRA Rating Model Development Guidelines*.

Ross, S. (2010). *Introduction to Probability Models* (10th ed.). Academic Press.

Siddiqi, N. (2006). *Credit Risk Scorecards*. John Wiley & Sons, Inc.

Sobehart, J., & Keenan, S. (2007, September). Understanding performance measure for validating default risk models: a review of performance metrics. *Journal of Risk Model Validation, 1*(2), 61-78.

Standard & Poor's. (2011). *Guide to credit rating essentials*.

Standard Chartered. (2008). *Shadow Bond Approach through Large Corp Scorecard (LCS)*. Presentation.

Taboga, M. (2012). *Lectures on Probability Theory and Mathematical Statistics*. Amazon CreateSpace.

Theil, H. (1971). *Principles of Econometrics*. John Wiley & Sons, Inc.

Trefethen, L. N., & Bau, D. (1997). *Numerical Linear Algebra*. Society for Industrial and Applied Mathematics.

Verbeek, M. (2004). *A Guide to Modern Econometrics* (2nd ed.). John Wiley & Sons Ltd.

Welch, B. L. (1947). The generalization of "Student's" problem when several different population variances are involved. *Biometrika*, 28-35.

Zijp, W. (1974). *Handleiding voor statistische toetsen*. H.D. Tjeenk Willink bv.

9 Appendices

A Histograms for the factor examples

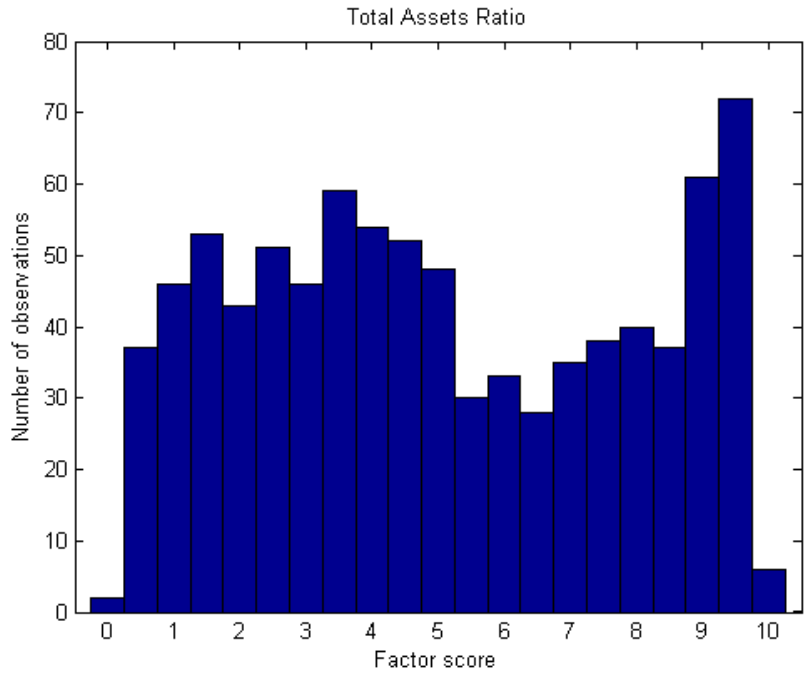

Figure 31: The histogram of the "total assets ratio" factor scores for the development dataset for the commercial banks model redevelopment.

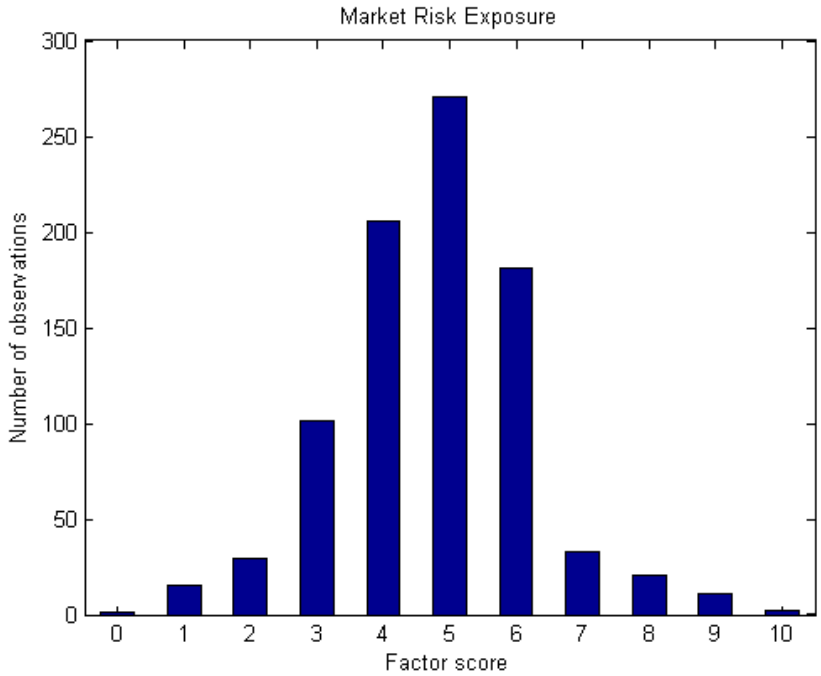

Figure 32: The histogram of the "market risk exposure" factor scores for the development dataset for the commercial banks model redevelopment.

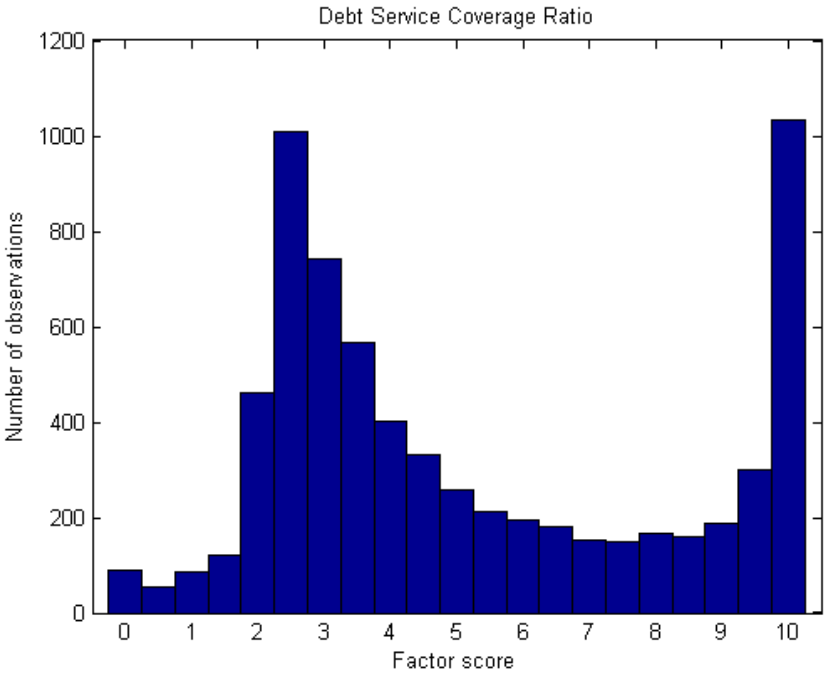

Figure 33: The histogram of the "debt service coverage ratio" factor scores for the development dataset for the Poland SME model redevelopment.

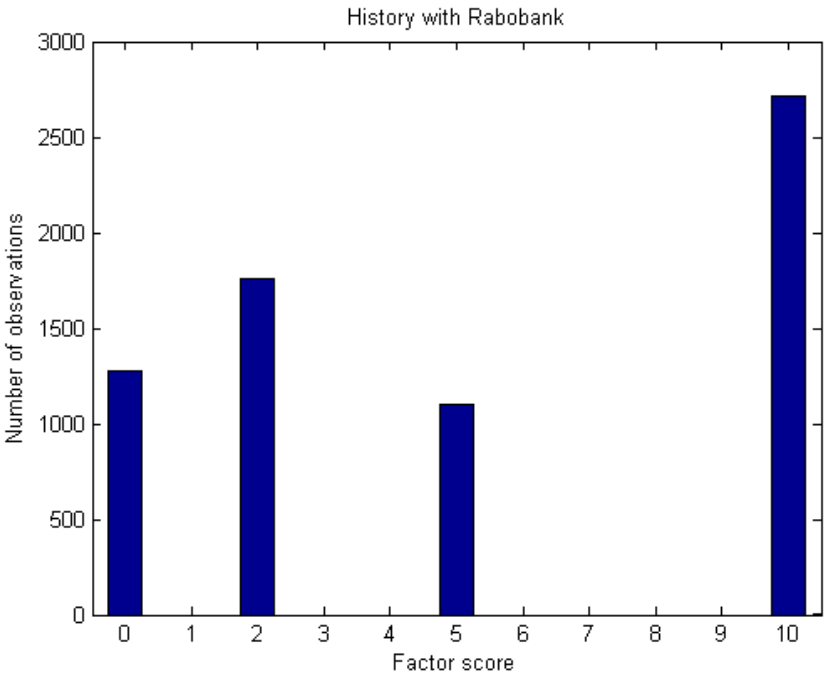

Figure 34: The histogram of the "history with Rabobank" factor scores for the development dataset for the Poland SME model redevelopment.

B The naive confidence region

If we have two estimated parameters, a naive confidence region could be formed by the rectangle with sides corresponding to the two one-dimensional confidence intervals. Suppose have two parameters, ξ_1 and ξ_2, whose one-dimensional confidence intervals (denoted by I_1 and I_2) have confidence levels α_1 and α_2, respectively. The rectangle can thus be defined by:

$$[(\xi_1, \xi_2) \mid \xi_1 \in I_1, \xi_2 \in I_2]$$

The probability of (ξ_1, ξ_2) lying in the rectangle is therefore given by $P(\xi_1 \in I_1, \xi_2 \in I_2)$. It then follows that:

$$\begin{aligned} P(\xi_1 \in I_1, \xi_2 \in I_2) &= 1 - P(\xi_1 \notin I_1, \xi_2 \in I_2) - P(\xi_1 \in I_1, \xi_2 \notin I_2) - P(\xi_1 \notin I_1, \xi_2 \notin I_2) \\ &\leq 1 - P(\xi_1 \notin I_1, \xi_2 \in I_2) - P(\xi_1 \notin I_1, \xi_2 \notin I_2) = 1 - P(\xi_1 \notin I_1) = \alpha_1 \end{aligned}$$

Similarly, we can derive that $P(\xi_1 \in I_1, \xi_2 \in I_2) \leq \alpha_2$.

If we set $\alpha_1 = \alpha_2$, it is clear that the rectangular region has a lower confidence level than α. For a confidence region with confidence level α, it should hold that the projection of this region onto the axes of ξ_1 and ξ_2 lies outside of the bounds defined by I_1 and I_2 for at least one of ξ_1 and ξ_2.

C The linear OLS estimator

In this appendix, we will show the derivation of the ordinary least squares estimator b^* for the linear model. In matrix notation, this model is given by:

$$Y = X\beta + \epsilon$$

If there are N observations and k explanatory factors, then Y is a $N \times 1$ vector, X a $N \times k$ matrix, β a $k \times 1$ vector, and ϵ a $N \times 1$ vector. Y contains the dependent variables, whereas X contains the independent variable values.

The β that minimizes $\epsilon^T\epsilon$ is the OLS estimator b^*. We can write:

$$\epsilon^T\epsilon = (Y - X\beta)^T(Y - X\beta) = Y^TY - 2\beta^TX^TY + \beta^TX^TX\beta$$

The derivative of $\epsilon^T\epsilon$ to β is then given by:

$$-2X^TY + 2X^TX\beta$$

Setting this derivative to zero, yields b^*. So:

$$b^* = (X^TX)^{-1}X^TY$$

This is the linear OLS estimator.

D Computing the tilt of an ellipse

We will now derive the expression for the angle of the tilt of an ellipse, denoted by θ. This is required for the analysis in 3.7.

Notice that for a tilted ellipse an alternative set of axes $(\tilde{x}, \tilde{y})$ can be defined, such that the ellipse is not tilted with respect to these axes. These alternative axes are therefore parallel to the major and minor axes of the ellipse. Therefore, θ is equal to the angle between the x axis and the $\tilde{x}$ axis. See Figure 35.

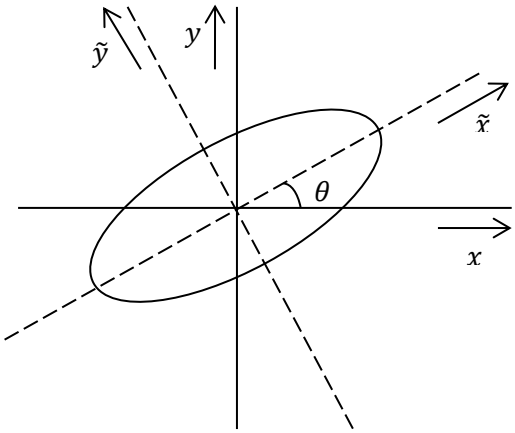

Figure 35: An ellipse with an alternative set of axes $(\tilde{x}, \tilde{y})$ parallel to the major and minor axes of the ellipse.

As relation between the two sets of axes can thus be seen as a standard rotation, the following equations hold:

$$\begin{cases} x = \tilde{x}\cos\theta - \tilde{y}\sin\theta \\ y = \tilde{x}\sin\theta + \tilde{y}\cos\theta \end{cases} \tag{29}$$

If we locate the centre of the ellipse at $(0,0)$ for simplicity, the ellipse equation in terms of x and y is given by:

$$ax^2 + bxy + cy^2 = 1 \tag{30}$$

The fixed coefficients for the equation are given by a, b, and c. By plugging (29) into (30), we obtain the equation in terms of $\tilde{x}$ and $\tilde{y}$.

$$\tilde{x}^2[a\cos^2\theta + b\cos\theta\sin\theta + c\sin^2\theta] + \tilde{x}\tilde{y}[(-a+c)\sin 2\theta + b\cos 2\theta] + \tilde{y}^2[a\sin^2\theta - b\cos\theta\sin\theta + c\cos^2\theta] = 1$$

Since the ellipse is not tilted with regards to the $(\tilde{x}, \tilde{y})$ axes, the $\tilde{x}\tilde{y}$ term should be zero. So:

$$(-a+c)\sin 2\theta + b\cos 2\theta = 0$$

From this, it follows that:

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{b}{a-c}\right)$$

The expression for the tilt of the ellipse can therefore be expressed in terms of a, b, and c.

E Matlab implementation of the ellipse method

The ellipse method is implemented in Matlab, a computational software program. Matlab is the program of choice for the statistical work on rating model (re)developments. Therefore, by implementing the ellipse method, it can be used directly during the next model (re)development.

A Matlab function is created which can create the inner and outer intervals for each factor coefficient in the model. Also, the ellipses can be plotted. The name of this function is `functionEllipseMethod`. Its input is the following:

- `v_dependent`: Vector containing the dependent variable for each observation. So this is either a binary variable for the logistic regression, or the logarithm of the PD corresponding to the external rating for linear regression.
- `m_data`: Matrix with the factor data for each observation. Each column corresponds to the factor scores of a factor that should be incorporated in the model.
- `s_method`: String indicating the regression method that should be used: `linear` for linear regression and `logit` for logistic regression.
- `s_intercept` (optional): String indicating whether an intercept should be used: `on` if yes, `off` if no.
- `plotplease` (optional): Variable indicating if a plot of all the ellipses should be made. If this is required, the variable should be set to 1, otherwise to 0 (default).
- `conflevel` (optional): Variable with the confidence level of the confidence region (ellipse). This variable can only have values within the range $(0,1)$. Its default value is 0.95.

The output of `functionEllipseMethod` are four matrices of the same size. They each have two columns (for the upper and lower boundaries) and the number of rows is equal to the number of factors in the model. The output consists of the following items:

- `int_in`: The matrix with the lower bound (first column) and upper bound (second column) of the inner interval for each factor coefficient.
- `int_out`: The matrix with the lower bound (first column) and upper bound (second column) of the outer interval for each factor coefficient.
- `int_in_wt`: The matrix with the lower bound (first column) and upper bound (second column) of the inner interval for each factor weight.
- `int_out_wt`: The matrix with the lower bound (first column) and upper bound (second column) of the outer interval for each factor weight.

The factor data that is used as input should only be the scores of factors that are allowed to be incorporated in the model. It may therefore be useful to first perform a stepwise regression on all factors to identify those that form the final model.

F Matlab implementation of the powerstat confidence interval

In order to facilitate the use of the confidence interval for the powerstat, the method to create this interval is implemented in Matlab. Matlab is the program used for developing the credit rating models, so it makes sense to implement the creation of the confidence interval in this program.

The Matlab function that is currently used for computing the powerstat is called `powerstatisticConform`. It is now extended under the new name `powerstatisticConform_ConfInt`. The input is not changed, but one output element is added: the confidence interval. The input arguments are:

- `v_score`: Vector with the model/factor scores.
- `v_PD`: Vector with the creditworthiness information (either binary or the logarithm of the PD).
- `s_plots` (optional): String that will be used as the title of the plot of the power curve, random curve and crystal ball curve. If this is empty or `no`, there will be no plot.
- `v_weights` (optional): Vector with the weight of each score. Default is a vector with ones.
- `s_logP` (optional): String indicating whether the logarithmic powerstat needs to be computed.

The output is the following:

- `p`: Value of the powerstat.
- `v_posNeg`: Vector with two entries: the percentages of the positive part (above the random curve) and negative part (below the random curve) of the power curve.
- `v_ConfInt`: Vector with two entries: the lower and upper bound of the confidence interval of the powerstat.

The confidence interval for the powerstat can only be created if the vector with creditworthiness information only contains 0's and 1's. If this is not the case, the interval returns `NaN`: not a number.

By default, a confidence level of 95% is used for the confidence interval. If required, this level can be changed directly in the code. In line 149 the variable `conflevel` can then be set to the required level of confidence.

G The empirical model score distributions

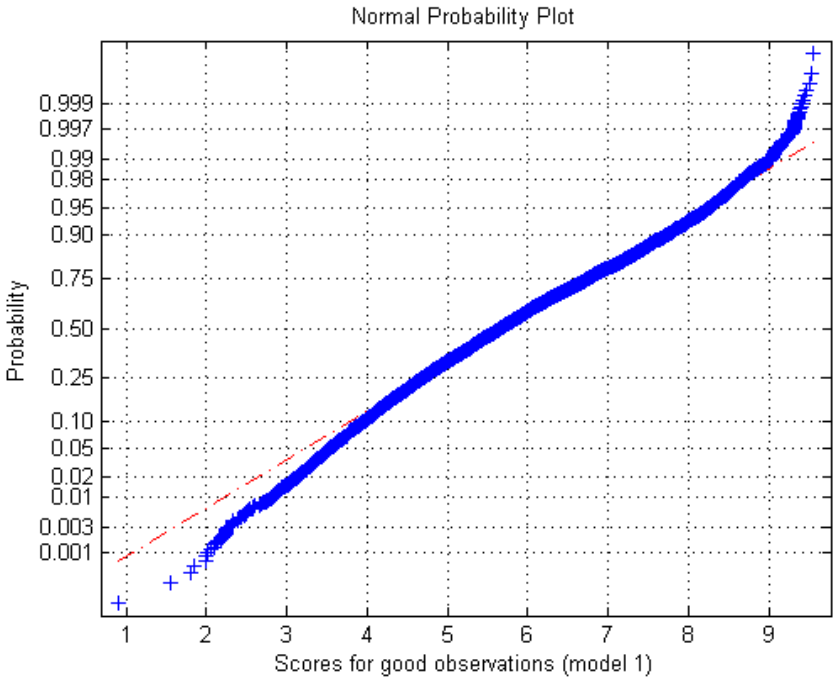

Figure 36: The empirical model score distribution together with the normal distribution with the same mean and standard deviation for the good observations in the development dataset of model 1.

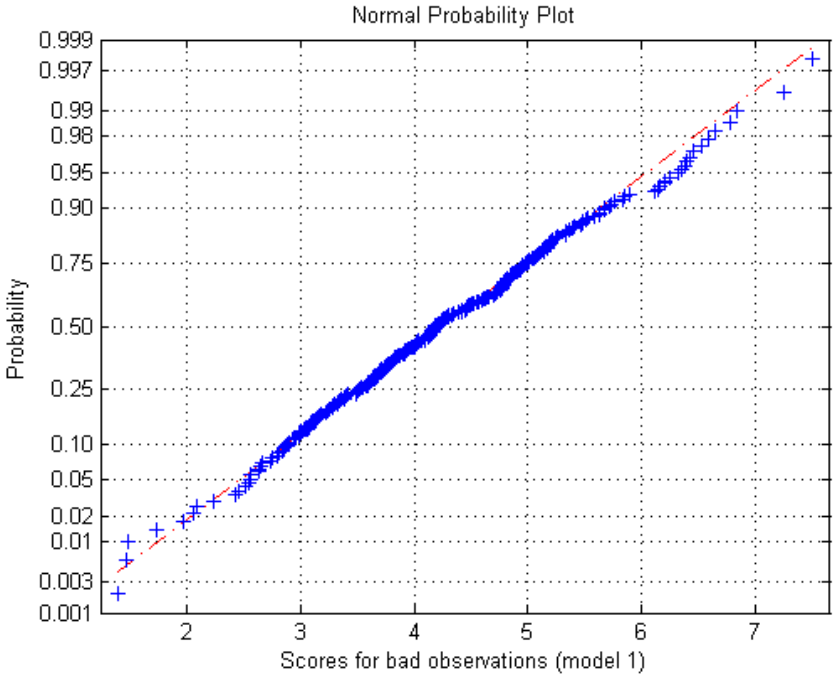

Figure 37: The empirical model score distribution together with the normal distribution with the same mean and standard deviation for the bad observations in the development dataset of model 1.

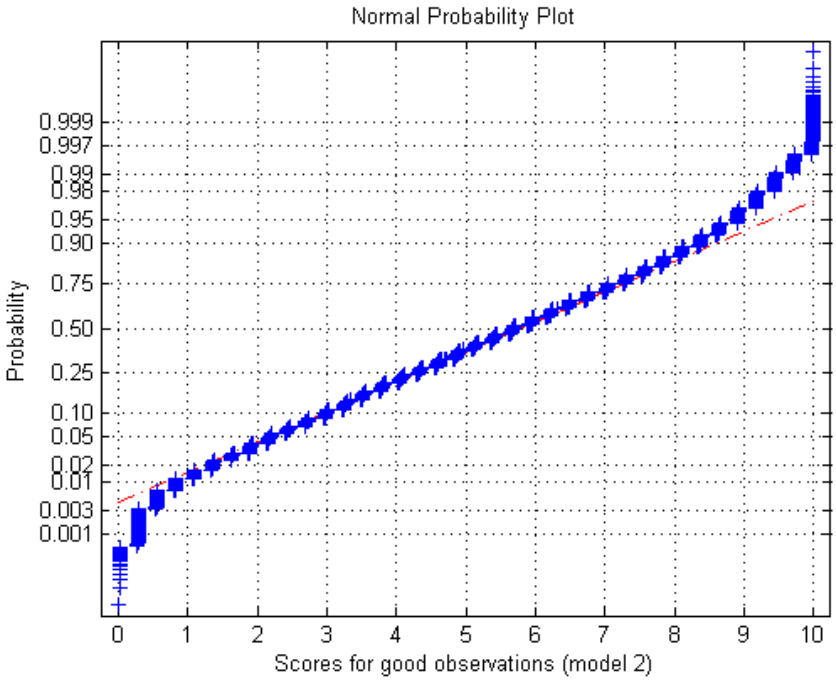

Figure 38: The empirical model score distribution together with the normal distribution with the same mean and standard deviation for the good observations in the development dataset of model 2.

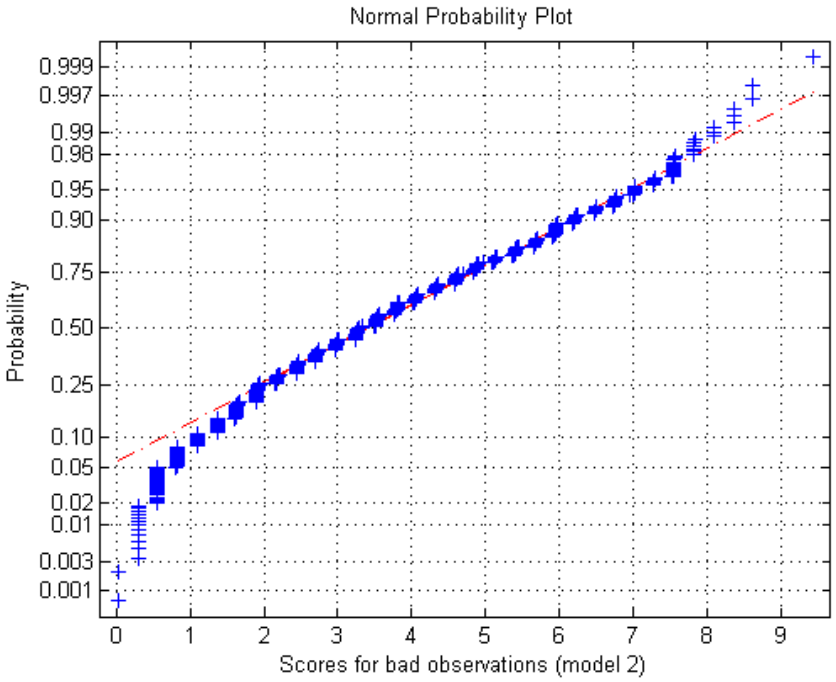

Figure 39: The empirical model score distribution together with the normal distribution with the same mean and standard deviation for the bad observations in the development dataset of model 2.

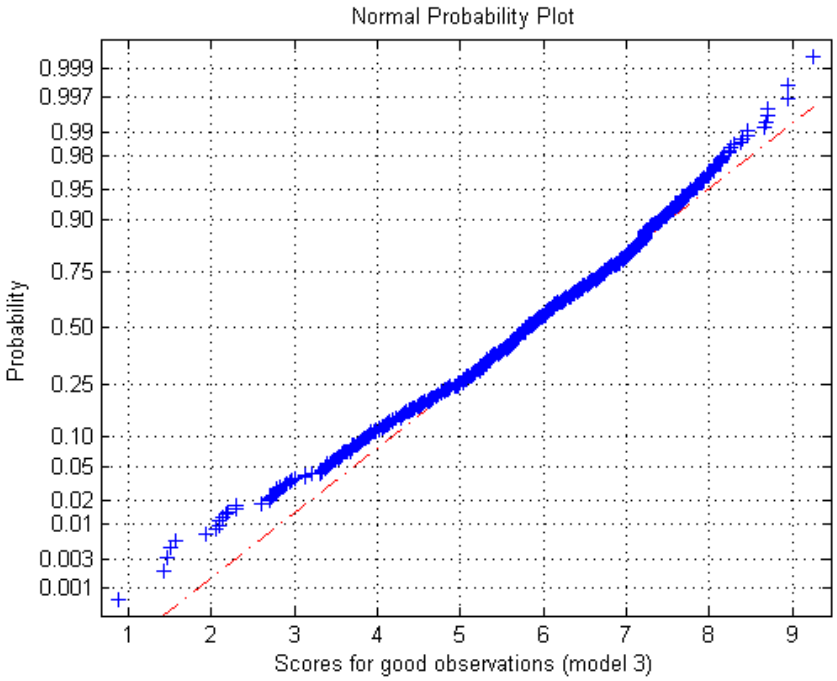

Figure 40: The empirical model score distribution together with the normal distribution with the same mean and standard deviation for the good observations in the development dataset of model 3.

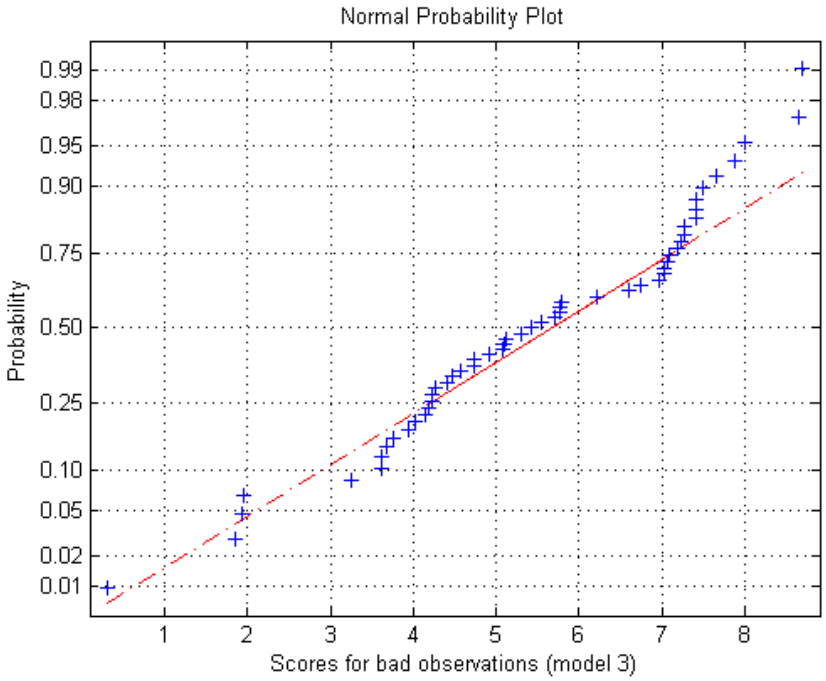

Figure 41: The empirical model score distribution together with the normal distribution with the same mean and standard deviation for the bad observations in the development dataset of model 3.